Copia riservata agli abbonati
ISSN 0392-8896

PERSONAL SOFTWARE

Spedizione in abbonamento postale gruppo III/70
ANNO 5 N. 36
FEBBRAIO 1986
L. 4.500

LA PRIMA RIVISTA EUROPEA DI SOFTWARE PER PERSONAL COMPUTER

GRUPPO EDITORIALE JACKSON
DIVISIONE PERIODICI



SKY VIDEOGAME PER SHARP

M20 PAINT

PRO.DOTTO PER APPLE

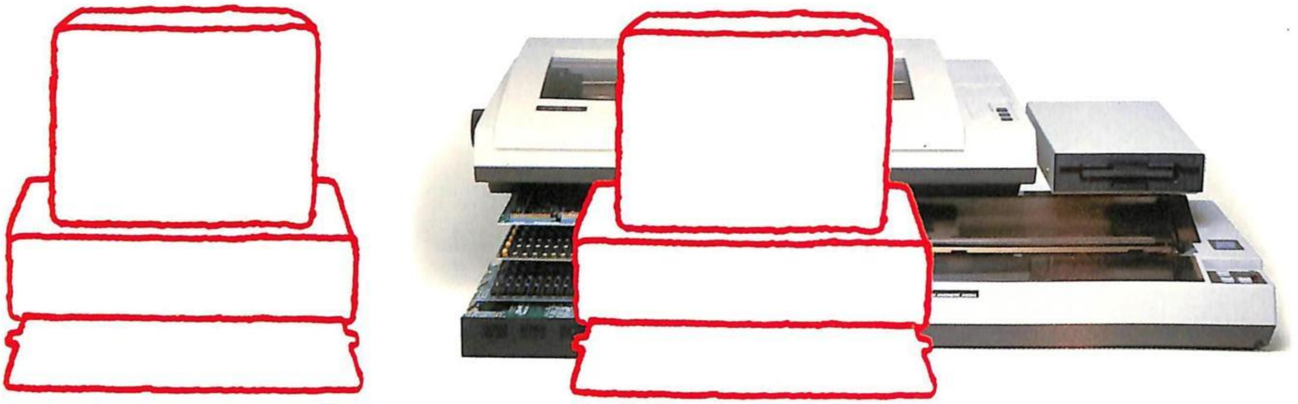
INPUT SIMULATO PER MSX

GRAFICA CON SINCLAIR QL

LIFE PER SINCLAIR QL

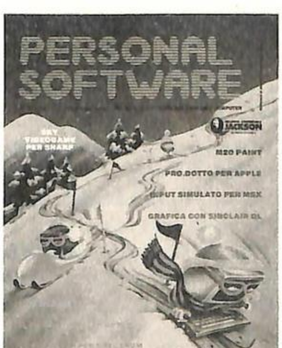
IL CAMPANARO PER APPLE

LUCY WRITER PER SPECTRUM



SOMMARIO

RUBRICHE



Questo mese:
Free sky videogame
per Sharp.

ANNO 5
N. 36
FEBBRAIO 1986

PERSONAL NEWS

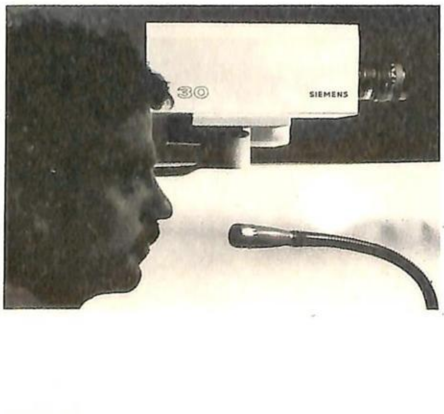
Giochi inglesi in riva al lago

Dalla fine del 1985 sono disponibili altre edizioni curate e tradotte di giochi di produzione inglese. La Lago di Como ha preso accordi con la Beyond/Monolith Software per distribuire in Italia alcuni dei più popolari titoli della Casa inglese. La confezione include le versioni del gioco su Spectrum e Commodore (ammesso che esistano entrambe) su un'unica cassetta, un manuale d'uso in italiano e la garanzia. I primi titoli: Quake Minus 1 (C 64), Enigmaforce e Bounces! (entrambi per C 64 e Spectrum 48 Kbyte), Superman in versione inglese (una cassetta per C 64 e Atari 400/800, un'altra per Spectrum 48 Kbyte e Amstrad Cpc 464).

Lago S.n.c.
Viale Massenzio Masia, 79
22100 Como
Tel. 031-552276

Come ti piloto la telecamera

La Siemens ha realizzato una telecamera pilotata dalla voce umana. Le diverse funzioni di comando (focale, fuoco, inclinazione, orientamento, ...) vengono eseguite



impartendo istruzioni con una parola prestabilita per ogni scelta. Gli ordini vengono trasmessi al sistema tramite un microfono, senza altri interventi meccanici. Il sistema, capace di riconoscere le sequenze di comandi dopo aver memorizzato le caratteristiche vocali dell'operatore, è particolarmente adatto per gli handicappati e per chi deve avere le mani libere per svolgere altre attività o perché opera in condizioni particolari.

Siemens Elettra S.p.A.
Via Fabio Filza, 25/A
20100 Milano
Tel. 67661

Utenti Lumena udite udite!

Rgb Computer Graphics, distributore esclusivo per l'Italia del software per la grafica pittorica Lumena II, offre a tutti gli utenti della prima release del prodotto uno scambio alla pari. Rgb su richiesta ritirerà il pacchetto completo del software Lumena e in cambio consegnerà il manuale e i dischetti relativi all'ultima release Lumena II. In questo modo scatta anche l'iscrizione automatica al Lumena Club che permette di entrare in contatto con altri utenti del pacchetto e dà diritto a una tessera-sconto del 10% su tutti i pacchetti optional di Lumena II. Rgb, società del gruppo Vtr, informa inoltre di aver messo a punto un sistema grafico per la trasmissione della Rai Disco Ring basata sulla scheda Pc Color 640, sull'interfaccia Cis Pal 640 e su un software ad hoc sviluppato a partire dalle "subroutine Halo", prodotti che la Casa milanese distribuisce in esclusiva su tutto il territorio nazionale. Il sistema prevede l'organizzazione della parte grafica che compare sul tabellone e la realizzazione dei simboli che definiscono la posizione in classifica dei vari motivi musicali.

Rgb Computer Graphics S.r.l.
Viale Piave, 17
20129 Milano
Tel. 02-783992

Memorex rinnova

Dopo aver allargato la propria penetrazione sul mercato italiano ed europeo degli utenti finali di dischetti per computer, la Memorex accompagna l'annuncio di una nuova strategia commerciale con l'introduzione di un nuovo marchio che caratterizzerà tutti i suoi prodotti magnetici flessibili. Il tutto per precisare l'immagine del prodotto e della marca di fronte a un mercato sempre più affollato e a un consu-



matore evoluto che ha bisogno di associare con facilità e immediatezza il marchio e le prestazioni. Insieme al nuovo look, la Memorex ha annunciato l'ingresso del suo ultimo prodotto: la versione a doppia faccia del microfloppy da 3,5" (in uso sulle più recenti generazioni di personal portatili: DG/One, Pro-Lite, Grid, Apricot e via discorrendo). L'alta coercitività (630 Oersted) consente di memorizzare sino a 500 Kbyte per facciata, per un totale di un milione di byte.

Memorex Italia
Via Ciro Menotti, 14
20129 Milano
Tel. 02-7491061

512 Kbyte di Ram per il QL

La Spem di Torino ha messo a punto un sistema per espandere la memoria interna del QL da 128 a 512 Kbyte. L'operazione è effettuata direttamente dalla Spem che inserisce le nuove Ram da 256 Kbyte e aggiunge un circuito per la commutazione degli indirizzi. Il servizio viene effettuato entro sette giorni con spedizione per posta contro assegno per 400.000 lire. Per i QL modificati la Spem fornisce una garanzia di tre mesi a esclusione dei microdrive.

Spem
Via Ponchielli, 26/C
10154 Torino
Tel. 011-856519

Un po' di tutto...

Fiere

• A Bologna, dal 22 al 26 Febbraio 1986, si terrà la quarta edizione del **Sioa**, Salone dell'informatica, della telematica e dell'organizzazione aziendale. Nell'ambito dell'esposizione verrà dato vita al "programma utenza", iniziativa che vuole fornire ai visitatori del Salone la mappa delle proposte finalizzate alle loro esigenze. - Consorzio Sioa - Via Napoli, 20 - Bologna.

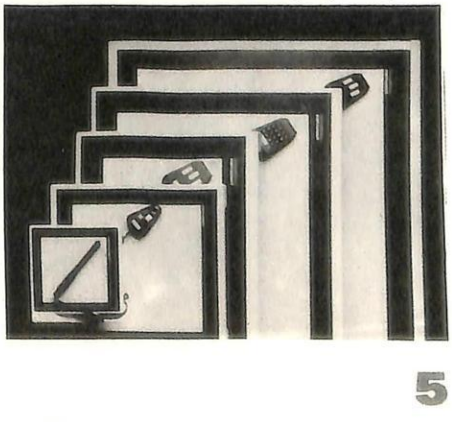
• La 20ma edizione del **Bias** (Mostra internazionale dell'automazione, strumentazione e microelettronica) si svolgerà a Milano dal 18 al 22 Marzo 1986. All'interno della mostra sono previste numerose manifestazioni culturali, scientifiche e tecniche. - Eiom - Viale Premuda, 2 - Milano.

• La **Mouse** S.r.l., società costituitasi nell'autunno dell'85 per la distribuzione di elettronica per l'ufficio, la scuola e l'hobby, annuncia l'apertura di un punto vendita a Seregno. Molte le marche rappresentate (Olivetti, Ibm, Canon, Toshiba, Commodore, Sinclair, Atari, Philips, Sharp) e vasta la biblioteca software a disposizione di chi vuole orientarsi nella selva di prodotti e pacchetti. - Mouse S.r.l. - Via Volta, 11 - Seregno (MI) - Tel. 0362-388275.

• Nella fabbrica **Ncr** di Dundee (UK) il noioso compito di verificare la robustezza dei terminali self service bancari (tipo le postazioni Bancomat) è stato affidato con successo a un robot Unimate Puma 5000 che inserisce la tesserina di plastica, preme i tasti per richiedere i contanti o l'estratto conto e ritira la tessera e le banconote al termine delle transazioni. Pare che l'avere tanti soldini per le "mani" non gli faccia poi un grande effetto... Ncr Italia - Viale Cassala, 22 - Milano - Tel. 02-838741.

Ce n'è per tutti

La db di Milano, rappresentante esclusivo in Italia della Houston Instruments, ha iniziato a distribuire sette nuovi modelli di digitalizzatori che possono soddisfare le più diverse esigenze di chi ha a che fare con la grafica. La serie True Grid è caratterizzata da risoluzione di 0,127 mm, precisione di 0,4 mm, scelta di trasduttori tra cursore a uno o quattro pulsanti e stilo (scrivente o non scrivente) e comprende tre modelli (13 per 13 cm, 28 per 28 cm, 28 per 43 cm). Per le applicazioni meccaniche, elettriche, architettoniche e di rilevamenti topografici e cartografici la Houston Instruments propone la gamma True Grid 8000. I quattro modelli (da 28 per 28 a 61 per 91 cm) hanno una risoluzione di 0,025 mm, una



precisione di 0,25 mm e permettono di scegliere diversi tipi di trasduttori, tra i quali spicca un cursore a 16 pulsanti, progettati esclusivamente per questa serie. Tutti sono dotati di interfaccia seriale Rs-232C.

db Electronics Instruments S.r.l.
Via Teano, 2
20161 Milano
Tel. 02-6469341

Trasportabile compatibile

La Mitac ha presentato il suo Viso al Taiwan Electronic Show e ce ne ha inviato notizia, essendo giustamente orgogliosa del suo ultimo prodotto. È un sistema compatto e completo (simile esteticamente all'Apple IIc) che non sacrifica alle ridotte dimensioni la componentistica. Sulla piastra madre monta infatti un 8088 a 4,77 MHz, 16 Kbyte di Rom, 512 Kbyte di Ram, il controller per il video grafico a colori, due porte seriali Rs-232C, un'uscita parallela compatibile Centronics, un orologio con batteria tampone e un connettore per un bus di espansione esterno. A differenza di molti portatili o trasportabili, il Viso non monta un drive da 3,5", ma incorpora un lettore di floppy da 5,25", permettendo così un uso immediato di tutti i package già distribuiti su supporti di questo formato. La possibilità di "personalizzare" il sistema lo rende appetibile anche a Oem e Var, mentre il prezzo assai competitivo dovrebbe fare la sua fortuna anche presso l'utente finale.



Mitac Inc.
585 Ming Sheng E. Rd
Taipei Taiwan ROC
Tel. (02)-5018231

Microelettronica a Vicenza

La rassegna dell'elettronica industriale e delle applicazioni informatiche che si è tenuta presso la fiera di Vicenza ha rappresentato un momento altamente qualificato per la piccola e media industria nel triangolo Vicenza, Treviso e Padova. Particolarmente significativa la presenza di giovani e operatori aziendali anche alle iniziative collaterali (convegni e incontri). Tra essi si sono distinti Msx Italia, il primo incontro nazionale dedicato ai micro del nuovo standard nippo-americano e Second-Source, uno spazio attrezzato per gli artigiani della mente (per chi ha inventato o costruito hardware o software fuori dal grande giro). La mostra svoltasi a Ottobre si ripeterà l'autunno prossimo.



Ente Fiera di Vicenza
Viale degli Scaligeri, 34
36100 Vicenza
Tel. 0444-96911

Club per Msx e Sinclair

Il Computer Club Roma-Talenti, già noto ai nostri lettori, conta quasi un centinaio di soci e ha deciso di estendere i suoi interessi anche al recente standard Msx, senza tralasciare il primo amore, lo Spectrum. La quota di iscrizione annuale è di 8.000 lire e dà diritto a un bollettino periodico oltre alla consultazione della biblioteca di libri e riviste del Club. Altra interessante iniziativa è la softeca con possibilità di scambio di programmi dei soci dietro semplice rimborso delle spese vive.

Computer Club Roma-Talenti
c/o Massimo D'Ascenzo
Via F. D'Ovidio, 109
00137 Roma
Tel. 06-8280043

Fuga al buio

Cimentatevi in questo gioco e subito vi accorgerete di non essere nel solito labirinto

di Ugo Rossini

Premessa

Quante volte vi siete cimentati in giochi che si svolgono in labirinti più o meno complessi? Parecchie volte, vero? Bene! Eccovene un altro! Però con alcune particolarità quanto meno originali.

Il gioco

Naturalmente dovrete cercare di uscire da una serie di labirinti, senza però toccare né le pareti, né gli ostacoli, né il Wafu (piccolo essere poco intelligente, ma pericolosissimo, che ha la cattiva abitudine di venirvi incontro).

Il problema principale è che il programma, dopo aver costruito il labirinto sotto i vostri occhi con l'uscita indicata da una grande U, farà scomparire il tutto lasciando visibili solamente l'omino (ovvero il vostro alter-ego) e il Wafu (di cui conoscete sempre la posizione grazie (?) al fortissimo odore che emana).

Nella parte inferiore dello schermo sono riportati, nell'ordine:

Radar: vi consente di visualizzare per un istante eventuali ostacoli, o il bordo, o l'uscita nella direzione in cui viene inviato il segnale. Ha una gittata ovviamente illimitata (si interrompe quando trova un ostacolo, segnalandolo). È mantenuto in efficienza da batterie portatili che si ricaricano continuamente; più velocemente ai livelli bassi e più lentamente a quelli alti. Non funziona se non ha a disposizione almeno l'11% della potenza totale;

Laser: serve per uccidere il Wafu, a patto che esso si trovi a non più di cinque unità di distanza da te. Purtroppo, come

Listato 1 - Il programma Fuga al buio.

```
10   REM  FUGA AL BUIO
20   PRINT  CHR$ (4)"BLOAD FAB.SHA
      PE,A$6000"
30   POKE 232, PEEK (43634): POKE
      233, PEEK (43635)
40   LOMEM: 24900
50   DIM A%(20,20)
60   HOME : PRINT "FUGA AL BUIO <<
      <<<<<<<<<>>>>>>>>>BY UGO"
70   PRINT : PRINT : INPUT "QUALE
      E' IL TUO NOME ";NO$
80 RA = 100:LA = 15:BR = 9:LI = 1
      :AU = 5:PU = 0:SE = 300
90   HGR2 : HCOLOR= 3: SCALE= 1: ROT=
      0
100  REM  COSTRUZIONE LABIRINTO
110  FOR K = 1 TO 20
120 A%(K,1) = 5:A%(K,20) = 5:A%(1
      ,K) = 5:A%(20,K) = 5
130  DRAW 5 AT (K + 7) * 8,4: DRAW
      5 AT (K + 7) * 8,156: DRAW 5
      AT 64,K * 8 - 4: DRAW 5 AT
      216,K * 8 - 4
140  NEXT
150  FOR K = 2 TO 19: FOR KK = 2 TO
      19:A%(K,KK) = 0: NEXT : NEXT
160  FOR K = 2 TO 19
170 L =  INT (15 *  RND (1) + 2)
180  FOR M = 1 TO L:N =  INT (18 *
      RND (1) + 2)
190 A%(K,N) = 1
200  DRAW 1 AT (K + 7) * 8,N * 8 -
      4
210  NEXT : NEXT
220 K =  INT (18 *  RND (1) + 2):
      N =  INT (18 *  RND (1) + 2)
230  IF A%(K,N) = 1 THEN  XDRAW 1
      AT (K + 7) * 8,N * 8 - 4
240 A%(K,N) = 4
250  DRAW 4 AT (K + 7) * 8,N * 8 -
      4
260  FOR SU = 1 TO 200:X =  PEEK
      ( - 16336): NEXT
270  HOME : HGR
280  GOSUB 1000: GOSUB 1470: GOSUB
      1100: GOTO 3000
1000  REM  TABELLA
1010  IF RA > 100 THEN RA = 100
```

Seguito listato Fuga al buio.

```
1020  VTAB 21: HTAB 1: PRINT "RAD
     AR ";RA; TAB( 12);"LEVEL ";L
     I; TAB( 24);">>SEC ";SE: PRINT
     "LASER ";LA; TAB( 12);"AUREA
      ";AU; TAB( 24);"POINT ";PU:
      PRINT "BREAK ";BR; TAB( 12)
     ;"MIVIS ";VI; TAB( 24);">>>M
     OV<<<"
1030  RETURN
1100  REM  PARTENZA SNAFU
1120 KK =  INT (18 *  RND (1) + 2
     ):NN =  INT (18 *  RND (1) +
     2): IF A%(KK,NN) = 1 OR A%(K
     K,NN) = 3 OR A%(KK,NN) = 4 THEN
     1120
1130  DRAW 2 AT (KK + 7) * 8,NN *
     8 - 4
1150  RETURN
1200  REM  AGGIORNAMENTO VALORI
1210 RB = RB +  RND (1) / LI: IF
     RB < 1 THEN 1230
1220 RA = RA + 1:RB = 0: IF RA <
     101 THEN  VTAB 21: HTAB 7: PRINT
     RA;"  "
1230 SF = SF + 1: IF SF < 4 THEN
     1260
1240 SF = 0:SE = SE - 1: VTAB 21:
      HTAB 30: PRINT SE"  ": IF S
     E = 0 THEN SE = 1
1250 VJ = VJ +  RND (1): IF VJ <
     1 THEN  RETURN
1260 VI = VI + 1:VJ = 0: IF VI =
     100 THEN VI = 0: FOR J = 1 TO
     200 / LI: POKE  - 16299,0: POKE
      - 16300,0:SU =  PEEK ( - 16
     336):SU =  PEEK ( - 16336): NEXT

1270  VTAB 23: HTAB 18: PRINT VI;
     "  "
1280  RETURN
1300  REM  DIREZIONE
1310 A =  PEEK ( - 16384):ZZ =  PEEK
     ( - 16368)
1320  GOSUB 1210: GOSUB 2000
1330  IF A = 213 OR A = 201 OR A =
     207 OR A = 208 THEN B = A: POP
     : GOTO 3020
1340  IF A = 215 THEN X = 0:Y =  -
     1: RETURN
1350  IF A = 193 THEN X =  - 1:Y =
     0: RETURN
1360  IF A = 196 THEN X = 1:Y = 0
     : RETURN
1370  IF A = 216 THEN X = 0:Y = 1
     : RETURN
1380  GOTO 1300
1400  REM  AUREA
1410  XDRAW 3 AT (K1 + 7) * 8,N1 *
     8 - 4
1420  FOR J = 1 TO 6
1430  XDRAW 9 AT (K + 7) * 8,N *
     8 - 4
1440  XDRAW 9 AT (K + 7) * 8,N *
     8 - 4: NEXT
1450 AU = AU - 1: IF AU =  - 1 THEN
     1600
1460  VTAB 22: HTAB 18: PRINT AU
1470 K =  INT (18 *  RND (1) + 2)
     :N =  INT (18 *  RND (1) + 2
     )
1480  IF A%(K,N) = 4 OR A%(K,N) =
     1 THEN 1470
1490 A%(K,N) = 3
1500  DRAW 3 AT (K + 7) * 8,N * 8
     - 4
1510 K1 = K:N1 = N
1520  RETURN
1600  REM MORTE
1610 F$ = "LA TUA AUREA PROTETTIV
     A E' STATA         ANNIENTATA
      E TU SEI STATO DILANIATO
        DALL'ULTIMA ESPLOSIONE..."
     : GOTO 1720
1700  REM USCITA DISTRUTTA
1710 F$ = "HAI COLPITO IN PIENO L
     'UNICA USCITA      ESISTENTE.
     .. COMPLIMENTI!!!"
1720  TEXT : HOME : VTAB 10: PRINT
     F$: VTAB 23: PRINT "PREMI UN
      TASTO PER CONTINUARE ";: GET
     R$: PRINT R$
1725  POKE  - 16384,0: GOSUB 4000

1730  VTAB 20: PRINT "COMUNQUE SI
     A: PREMI -RETURN- PER
      RIPARTIRE OPPURE UN ALTRO T
     ASTO PER      TERMINARE"
1750  GET RI$: PRINT RI$
1760  IF RI$ =  CHR$ (13) THEN  RUN
     40
1770  END
1800  REM  LIVELLO
1810  TEXT : HOME : VTAB 10: HTAB
     4: PRINT "SEI USCITO DAL ";L
     I;"' LIVELLO E SEI"
1820  HTAB 7: PRINT "ENTRATO NEL
     ";LI + 1;"' LABIRINTO."
1830  IF LI >  = 8 THEN 1850
1840 AA$ = AA$ + "000":A$ = "ATTE
     NT" + AA$
```

insegna la saggezza popolare: "Morto un Wafu, se ne fa un altro". Infatti, arriverà subito un amico dell'ucciso ben deciso a compiere (giusta) vendetta. Il laser scalfisce soltanto ostacoli, bordo e uscita, senza però distruggerli. All'inizio del gioco si ha una dotazione di 15 colpi laser;

Break: è un'arma potente con gittata breve (tre unità). Distrugge la prima cosa che si trova sulla sua strada, disintegrandola. Utilissima contro gli ostacoli (e contro il Wafu). Ma fate attenzione: non dovete *mai* colpire il bordo di recinzione con un Break. Infatti, dovete sapere che tutto l'insieme dei labirinti è scavato in un vulcano (attivo). Vi lascio immaginare le conseguenze (drammatiche) nel caso venisse sfondata la barriera che separa voi dal magma infuocato. Inoltre dovrete anche evitare accuratamente di colpire l'unica uscita esistente! Avete a disposizione nove Break;

Level: vi dà l'indicazione del livello in cui vi trovate;

Aurea: rappresenta il vostro... stato di salute. La tuta che vi riveste, infatti, ha la capacità di assorbire cinque assalti prima di cedere e porre così fine alla vostra presenza (nel gioco);

Mivis: non è facile spiegare il Mivis (visione mistica). Un'ipotesi avanzata, da un gruppo di studiosi, lo fa risalire a particolari elementi radioattivi presenti nelle rocce vulcaniche, ma di sicuro non c'è niente.

Il fatto è che si sviluppa gradualmente in voi una capacità di visualizzare (nel momento in cui raggiunge l'apice, ovvero 100) per qualche secondo tutto il labirinto con la vostra posizione, quella del Wafu, e soprattutto l'uscita.

Ai livelli bassi la visione dura maggiormente, mentre tende a scemare con il susseguirsi dei livelli;

Sec: scandisce il passare del tempo (a ritroso), e serve unicamente a calcolare, una volta raggiunta l'uscita, il punteggio (vedi Point).

All'inizio di ogni livello avrete 300 secondi a disposizione;

Point: il punteggio ottenuto. I punti vengono così assegnati: 200 punti per ogni Wafu distrutto, 100 punti per ogni livello superato più un punto per ogni secondo rimasto al raggiungimento dell'uscita;

Movim: è l'indicatore del movimento. Esso, come anche Radar, Break e Mivis, si accenderà (leggi inverse) quando è in funzione (vedi Comandi).

Ogni volta che si raggiunge l'uscita si ha diritto a una dotazione supplementare (a patto che non si superi la dotazione iniziale):

- 30 unità Radar;
- 2 unità Laser;
- 1 unità Break.

I comandi

È consigliabile usare tutte e due le mani (ma anche coi piedi può essere divertente: provare per credere...): sarà un ottimo esercizio di coordinazione.

È vero che la maggior parte delle volte le decisioni potranno essere prese con tutta calma (anche se i secondi, e quindi i punti, continueranno a diminuire).

Seguito listato Fuga al buio.

```
1850  PRINT : PRINT : HTAB (19 -
      LEN (A$) / 2): PRINT A$ + "
      !!!"
1860  FOR J = 1 TO 5000: NEXT : PRINT
      :LI = LI + 1
1870 RA = RA + 30: IF RA > 100 THEN
      RA = 100
1880 LA = LA + 2: IF LA > 15 THEN
      LA = 15
1890 BR = BR + 1: IF BR > 9 THEN
      BR = 9
1900 PU = PU + SE + 100
1910 SE = 300
1920  GOTO 90
2000  REM  SNAFU
2010 RI = RI + 1: IF RI < 4 THEN
      RETURN
2020 RI = 0
2030 KB = KK - (KK > K) + (KK < K
      ):NB = NN - (NN > N) + (NN <
      N)
2040  IF A%(KB,NB) = 1 AND  RND (
      1) < .02 * LI THEN 2070
2060  IF A%(KB,NB) = 5 OR A%(KB,N
      B) = 1 OR A%(KB,NB) = 4 THEN
      RETURN
2070  XDRAW 2 AT (KK + 7) * 8,NN *
      8 - 4
2080  IF A%(KB,NB) = 3 THEN  GOSUB
      1400
2090 A%(KK,NN) = 0
2100  DRAW 2 AT (KB + 7) * 8,NB *
      8 - 4
2110 A%(KB,NB) = 2
2120 KK = KB:NN = NB
2130  RETURN
3000  REM  MAIN
3010 B =  PEEK ( - 16384):BB =  PEEK
      ( - 16368)
3020  IF B = 213 THEN  GOSUB 1000
      : VTAB 23: HTAB 24: INVERSE
      : PRINT ">>>MOV<<<";: NORMAL
      : GOSUB 1300: GOTO 3100
3030  IF B = 201 THEN  GOSUB 1000
      : VTAB 21: HTAB 1: INVERSE :
      PRINT "RADAR": NORMAL : GOSUB
      1300: GOTO 3210
3040  IF B = 207 THEN  GOSUB 1000
      : VTAB 22: HTAB 1: INVERSE :
      PRINT "LASER": NORMAL : GOSUB
      1300: GOTO 3410
3050  IF B = 208 THEN  GOSUB 1000
```

ma quando il Wafu si muove è meglio agire con prontezza prima che vi raggiunga.

Mano destra

U = Movimento;
I = Radar;
O = Laser;
P = Break.

Una volta premuto uno di questi tasti (disposti uno dopo l'altro e raggiungibili quindi da quattro dita diverse), la funzione corrispondente apparirà in inverse, confermandovi così di essere pronta ad accettare la direzione.

Mano sinistra

W = Nord;
X = Sud;
A = Ovest;
D = Est.

Quindi, a esempio, la giusta sequenza nel caso vogliamo dirigerci a Nord è: U W; mentre se vogliamo sbarazzarci di un ostacolo a Ovest utilizzeremo il Break premendo in successione: P A. Sembra complesso, ma una volta che vi sarete impratichiti, vi risulterà molto semplice e veloce, e non vi costringerà soprattutto a guardare la tastiera continuamente.

Caricamento del programma

Scrivete la tavola delle figure (Fab.Shape) e il programma Fuga al buio. La prima volta, inoltre, date il Run anche al programma Creaz Fab. Rec che creerà un file (Fab.Rec) utilizzato dal programma principale per registrare i dieci migliori risultati di tutti i tempi.

Conclusione e suggerimenti

1) Si possono sviluppare diverse tecniche per riuscire a sopravvivere più a lungo. L'importante, comunque, è di non farsi mai prendere dal panico e cercare di ricordarsi sempre la direzione in cui si trova l'uscita.

2) Cercate di usare i Break con parsimonia e solo quando ne avete realmente bisogno, perché è grazie a questa micidiale arma che sarete in grado di superare ogni record e di battere i vostri amici/concorrenti.

3) Il Wafu ha la particolare caratteristica di riuscire, a volte, a distruggere gli ostacoli che incontra. I Wafu dei livelli più alti sono più forti e quindi distruggono più facilmente i blocchi e si avvicinano quindi più velocemente alla vostra posizione.

4) La visione mistica agisce sulla vostra memoria e vi farà quindi vedere il labirinto come era all'inizio e non com'è al momento del Mivis. Quindi eventuali ostacoli distrutti da voi e/o dai Wafu figurano presenti anche se in realtà non esistono più.

5) Controllate sempre che la funzione desiderata sia accesa prima di premere un tasto direzionale: la fretta può portarvi a risultati disastrosi.

6) Se volete interrompere momentaneamente il gioco (leggi: telefono) premete Control-S.

7) E ora... Run! ■

Seguito listato Fuga al buio.

```
     : VTAB 23: HTAB 1: INVERSE :
      PRINT "BREAK": NORMAL : GOSUB
     1300: GOTO 3710
3060  GOSUB 1210: GOSUB 2000: GOTO
     3010
3100  REM  MOVIMENTO
3110 K = K + X:N = N + Y
3120  GOSUB 1210: GOSUB 2000
3130  IF A%(K,N) = 0 THEN  XDRAW
     3 AT (K1 + 7) * 8,N1 * 8 - 4
     :A%(K1,N1) = 0:SU =  PEEK ( -
     16336): DRAW 3 AT (K + 7) *
     8,N * 8 - 4:A%(K,N) = 3:K1 =
     K:N1 = N: GOSUB 1300: GOTO 3
     100
3140  IF A%(K,N) = 1 OR A%(K,N) =
     2 OR A%(K,N) = 5 THEN  GOSUB
     1400: GOSUB 1000: GOTO 3010
3150  IF A%(K,N) = 4 THEN 1800
3160  GOTO 3010
3200  REM  RADAR
3210 X1 = X:Y1 = Y
3220  IF RA < 11 THEN  GOSUB 1000
     : GOTO 3010
3230  GOSUB 1210: GOSUB 2000
3240 L = A%(K + X,N + Y)
3250  IF L = 0 THEN  DRAW 6 AT (K
     + 7 + X) * 8,(N + Y) * 8 -
     4:SU =  PEEK ( - 16336): XDRAW
     6 AT (K + 7 + X) * 8,(N + Y)
     * 8 - 4:X = X + X1:Y = Y +
     Y1: GOTO 3240
3280  IF L = 2 THEN 3320
3290  DRAW L AT (K + 7 + X) * 8,(
     N + Y) * 8 - 4
3300  FOR J = 1 TO 60: NEXT
3310  XDRAW L AT (K + 7 + X) * 8,
     (N + Y) * 8 - 4
3320 RA = RA - 5: VTAB 21: HTAB 7
     : PRINT (RA > 0) * RA;"  "
3330  GOSUB 1300: GOTO 3210
3400  REM  LASER
3410 X1 = X:Y1 = Y
3420  IF LA = 0 THEN  GOSUB 1000:
     GOTO 3010
3430 LA = LA - 1: VTAB 22: HTAB 7
     : PRINT LA;" "
3440  GOSUB 1210: GOSUB 2000
3450  FOR GI = 1 TO 5
3460 L = A%(K + X,N + Y)
3470  IF L = 0 THEN  DRAW 7 AT (K
     + 7 + X) * 8,(N + Y) * 8 -
```

APPLE II

Seguito listato Fuga al buio.

```
      4:SU =  PEEK ( - 16336): XDRAW
      7 AT (K + 7 + X) * 8,(N + Y)
       * 8 - 4:X = X + X1:Y = Y +
      Y1: NEXT : GOSUB 1300: GOTO
      3410
3500  IF A%(K + X,N + Y) = 2 THEN
      3550
3510  DRAW A%(K + X,N + Y) AT (K +
      7 + X) * 8,(N + Y) * 8 - 4
3520  FOR SV = 1 TO 10:SU =  PEEK
      ( - 16336): NEXT
3530  XDRAW A%(K + X,N + Y) AT (K
      + 7 + X) * 8,(N + Y) * 8 -
      4
3540  GOSUB 1300: GOTO 3410
3550  XDRAW 2 AT (K + 7 + X) * 8,
      (N + Y) * 8 - 4:A%(K + X,N +
      Y) = 0
3560  FOR J = 1 TO 3
3570  DRAW 9 AT (K + 7 + X) * 8,(
      N + Y) * 8 - 4
3580 SU =  PEEK ( - 16336)
3590  XDRAW 9 AT (K + 7 + X) * 8,
      (N + Y) * 8 - 4: NEXT
3600 PU = PU + 200: GOSUB 1000
3610  GOSUB 1100
3620  GOSUB 1300: GOTO 3410
3700  REM  BREAK
3710 X1 = X:Y1 = Y
3720  IF BR = 0 THEN  GOSUB 1000:
      GOTO 3010
3730 BR = BR - 1: VTAB 23: HTAB 7
     : PRINT BR
3740  GOSUB 1210: GOSUB 2000
3750  FOR GI = 1 TO 3
3760 L = A%(K + X,N + Y)
3770  IF L = 0 THEN  DRAW 8 AT (K
      + 7 + X) * 8,(N + Y) * 8 -
      4:SU =  PEEK ( - 16336): XDRAW
      8 AT (K + 7 + X) * 8,(N + Y)
       * 8 - 4:X = X + X1:Y = Y +
      Y1: NEXT : GOSUB 1300: GOTO
      3710
3800  DRAW L AT (K + 7 + X) * 8,(
      N + Y) * 8 - 4
3810  FOR SV = 1 TO 5:SU =  PEEK
      ( - 16336): NEXT
3820  XDRAW L AT (K + 7 + X) * 8,
      (N + Y) * 8 - 4
3830  FOR J = 1 TO 3
3840  DRAW 9 AT (K + 7 + X) * 8,(
      N + Y) * 8 - 4
3850 SU =  PEEK ( - 16336)
3860  XDRAW 9 AT (K + 7 + X) * 8,
      (N + Y) * 8 - 4: NEXT
3870  IF L = 1 OR L = 2 THEN A%(K
      + X,N + Y) = 0
3880  IF L = 2 THEN  GOSUB 1100:P
      U = PU + 200: GOSUB 1000
3890  IF L = 4 THEN 1700
3900  IF L = 5 THEN 3920
3910  GOSUB 1300: GOTO 3710
3920  FOR J = 1 TO 10: POKE  - 16
     300,0: PRINT  CHR$ (7): POKE
      - 16299,0:SU =  PEEK ( - 16
     336): NEXT
3930 F$ = "PECCATO...
TE L'AVEVO DETTO DI STARE ATTENTO
    ...
HAI SFONDATO LA PARETE ESTERNA
E IL MAGMA INFUOCATO CHE CIRCONDA

COMPLETAMENTE IL LABIRINTO
LO HA INVASO UCCIDENDOTI..."
3940  GOTO 1720
4000  REM   REGISTRAZIONE
4020  PRINT  CHR$ (4)"OPEN FAB.RE
     C"
4030  PRINT  CHR$ (4)"READ FAB.RE
     C"
4040  FOR J = 1 TO 10
4050  INPUT A$(J),A(J)
4060  NEXT
4070  FOR J = 1 TO 10
4080  IF A(J) < PU THEN 4110
4090  NEXT
4100  GOTO 4210
4110  FOR K = 9 TO J STEP  - 1
4120 A$(K + 1) = A$(K):A(K + 1) =
     A(K)
4130  NEXT
4140 A$(J) = NO$:A(J) = PU:J = 10

4150  PRINT  CHR$ (4)"DELETE FAB.
     REC"
4160  PRINT  CHR$ (4)"OPEN FAB.RE
     C"
4170  PRINT  CHR$ (4)"WRITE FAB.R
     EC"
4180  FOR J = 1 TO 10
4190  PRINT A$(J): PRINT A(J)
4200  NEXT
4210  PRINT  CHR$ (4)"CLOSE"
4215  HOME : INVERSE : PRINT "ECC
     O I DIECI MIGLIORI RISULTATI
     :.......": NORMAL
4220  FOR J = 1 TO 10
4230  VTAB J + 3: PRINT A$(J); TAB(
     25);A(J)
4240  NEXT
4250  RETURN
```

Pro.dotto

Una pro.fittevole serie di pro.grammi per lavorare con ProDos

di Luca Accomazzi *Parte prima*

Domanda: perché mai si dovrebbe programmare o lavorare sotto ProDos? Conosciamo il Dos 3.3 benissimo, e non siamo costretti a reimparare una marea di dettagli differenti!
Risposta: bisognerebbe imparare a interagire con ProDos per lo stesso motivo per cui viene usato un word processor anziché una macchina per scrivere: è più moderno, più veloce, sfrutta meglio le risorse disponibili. Il computer viene meglio valorizzato con ProDos che con Dos, la cui struttura è ancora essenzialmente quella pensata nel 1978 per degli Apple II, ben diversi dai modelli attuali.
Domanda: non sarebbe sufficiente utilizzare uno dei nuovi Dos 3.3 compatibili superveloci? Molti di questi affermano di poter rivaleggiare in velocità con lo stesso ProDos.
Risposta: i Dos 3.3 compatibili velocizzati hanno lo stesso assetto di una Fiat 500 con motore truccato per portarlo a 2.000 cc! Innanzitutto, la gran massa delle modifiche necessarie a rendere il 3.3 competitivo con ProDos - per esempio sostituire le routine che permettono di usare Applesoft in Ram per chi non ha né Applesoft in Rom né 64 Kbyte (cioè non servono a nessuno) oppure il comando Chain che funziona solo con l'obsoleto Integer Basic - non sono state compiute per nessun Dos velocizzato.

In secondo luogo, questi sono più veloci solo per i programmi in linguaggio macchina (file binari) e poco migliori dell'originale 3.3 per i programmi Basic; ma soprattutto, quasi tutti sono scritti male e male testati! Una prova lampante: tutti questi Dos modificati hanno apportato cambiamenti alla funzione Init, spesso sostituendola del tutto. Ma alle routine di Init fanno riferimento altre parti di Dos, cosicché ciascuna di queste tanto vantate migliorie è minata sin dalla nasciata.
Domanda: a parte la velocità, ProDos è migliore di Dos 3.3? Si è sentito dire che è anche pieno di bachi, errori e imperfezioni!
Risposta: sì, ProDos è superiore al 3.3 in molti aspetti oltre la velocità: permette l'uso di Profile e del disco Ram, il salvataggio di variabili Basic con il comando Store, il Chain per i programmi Applesoft e soprattutto è molto più semplice da usare da linguaggio macchina. Ben presto, nuovi programmi che utilizzano ProDos saranno in circolazione e saranno tanto più efficienti dei vecchi, nell'uso del disco tanto da rendere sciocco il solo confronto: è già disponibile il nuovo AppleWriter II 2.0 sotto ProDos, circa tre volte più veloce nella gestione dei testi e perfetto per l'uso su un IIc.

Certo, ProDos ha alcuni difetti di programmazione, tanto che ne sono state distribuite ben poche versioni al momento attuale. Ma ProDos è stato costruito in modo da mantenersi compatibile tra versioni differenti, al contrario di quanto accadde quando al Dos 3.2 venne sostituito il 3.3, e dunque potete sostituire il vecchio ProDos con una versione più recente e affidabile in ogni momento in cui venga resa disponibile, senza dover convertire dischi o file.
Domanda: ma ProDos è povero di programmi pronti che permettono di usarlo facilmente, programmi disponibili invece in gran quantità con il 3.3!
Risposta: vero. Ecco dove si inserisce Pro.dotto, una raccolta di applicazioni che lavorano sotto ProDos e che ne rendono facile l'uso, che aiutano a gestire i dischi, a programmare, ad apprendere correttamente l'uso del nuovo sistema operativo.

Introduzione al Pro.dotto

In questa puntata e nelle quattro che seguiranno mensilmente vi presenteremo un tool-kit, cioè una serie di attrezzi di lavoro, per sfruttare a fondo le potenzialità del sistema operativo ProDos sull'Apple II. Si tratta di una serie di programmi specifici, creati per soddisfare una serie di bisogni generici e utili in molteplici situazioni: suggeriremo strada facendo alcune possibili applicazioni. Chiameremo l'insieme dei quattro programmi Pro.dotto; si tratta comunque di quattro programmi distinti che condividono una serie di prerogative:

- sono interamente realizzati nel linguaggio macchina 6502 dell'Apple per sfruttarne a fondo le possibilità e la potenza;
- sono in grado di lavorare sotto una qualunque versione di ProDos, dalla 1.0 alla 1.1.1, ultima versione distribuita, e con ogni probabilità funzioneranno anche con le future versioni, finché la Apple non modificherà le caratteristiche generali di ProDos, dettagliate nel proprio manuale tecnico;
- sono in grado di interagire tra loro e risiedere contemporaneamente in memoria;
- funzionano su un Apple II+, IIe, IIc con almeno 64 Kbyte di memoria e un disk drive. L'unica richiesta aggiuntiva è la presenza, per uno dei quattro programmi, della scheda 80 colonne - incorporata nell'Apple IIc e disponibile per gli altri modelli.

I quattro programmi sono chiamati Pro.Menu, Editor di blocco, Help e Il padrone della pagina globale. Il secondo e il terzo programma sono stati sviluppati in più versioni che differiscono leggermente per la zona di memoria in cui risiedono e per certe caratteristiche aggiuntive che verranno descritte nelle puntate che li riguardano: i loro listati con i commenti saranno pubblicati nelle puntate seguenti, uno alla volta. Per motivi di spazio non potremo pubblicare i listati di ciascuna versione del programma: per poter disporre di

Listato 1.

```
SOURCE    FILE #01 =>FP
0000:                  1 *
0000:                  2 * FP ProDOS
0000:                  3 * Creato 30LUG85MAR
0000:                  4 * MisterAkko
0000:                  5 *
----- NEXT OBJECT FILE NAME IS FP.OBJ
0300:          0300    6            ORG     $0300
0300:                  7 *
0300:                  8 * EQUate table
0300:                  9 *
0300:          00D6   10 RUNFLAG    EQU     $D6
0300:          BEF8   11 FREBUF     EQU     $BEF8
0300:          D64B   12 SCRTCH     EQU     $D64B
0300:          0800   13 BASSTRT    EQU     $0800
0300:          BF59   14 BITMAP     EQU     $BF59
0300:          0074   15 MEMSIZ     EQU     $74             ;Himem
0300:          0070   16 FRETOP     EQU     $70             ;Inizio pool stringhe
0300:          0067   17 TXTTAB     EQU     $67
0300:          BE06   18 EXTRNCMD   EQU     $BE06
0300:          03F5   19 AMPSND     EQU     $03F5
0300:                 20 *
0300:                 21 * BEGIN
0300:                 22 *
0300:20 F8 BE         23 FP         JSR     FREBUF          ;Resetta MAXFILES
0303:A9 00            24            LDA     #0
0305:8D 00 08         25            STA     BASSTRT         ;New
0308:8D 01 08         26            STA     BASSTRT+1
030B:8D 02 08         27            STA     BASSTRT+2
030E:85 D6            28            STA     RUNFLAG         ;Azzera RUN_FLAG
0310:A2 00            29            LDX     #0
0312:                 30 *
0312:                 31 * Reset della PAGINA GLOBALE, protezione
0312:                 32 *
0312:9D 59 BF         33 LOOPH2     STA     BITMAP,X
0315:E8               34            INX
0316:E0 12            35            CPX     #18             ;Qui sproteggiamo da $800 a
 $9700!
0318:D0 F8    0312    36            BNE     LOOPH2
031A:A9 CF            37            LDA     #$CF            ;Protette le pag. 0,1 e 4-7

031C:8D 58 BF         38            STA     $BF58
031F:A9 3F            39            LDA     #$3F            ;Protette le pagine $9A-$9F

0321:8D 6B BF         40            STA     $BF6B
0324:A9 FF            41            LDA     #$FF
0326:A2 00            42            LDX     #0
0328:9D 6C BF         43 GPG        STA     $BF6C,X
032B:E8               44            INX
032C:E0 03            45            CPX     #$03
032E:D0 F8    0328    46            BNE     GPG
0330:A9 C3            47            LDA     #$C3
0332:8D 6F BF         48            STA     $BF6F           ;4 pagine non protette in f
ondo...
0335:                 49 *
0335:                 50 * Fine del reset della protezione in bit map
0335:                 51 *
0335:A9 96            52            LDA     #$96            ;Fissa HIMEM
0337:85 74            53            STA     MEMSIZ          ; lasciando 1K di buffer
0339:85 70            54            STA     FRETOP          ; e similmente il pool stri
nghe
033B:A9 01            55            LDA     #1
033D:85 67            56            STA     TXTTAB
033F:A9 08            57            LDA     #8
0341:85 68            58            STA     TXTTAB+1        ;Applesoft prgm start at $0
801
0343:A9 BE            59            LDA     #$BE
0345:8D 08 BE         60            STA     EXTRNCMD+2      ;Qui si resetta
0348:A9 9E            61            LDA     #$9E            ; EXTRNCMD
034A:8D 07 BE         62            STA     EXTRNCMD+1      ; per puntare a
034D:A9 4C            63            LDA     #$4C            ; XRETURN
034F:8D 06 BE         64            STA     EXTRNCMD
0352:A9 03            65            LDA     #3
0354:8D F6 03         66            STA     AMPSND+1        ;Qui si resetta &!
0357:A9 BE            67            LDA     #190
0359:8D F7 03         68            STA     AMPSND+2        ;Qui si resetta ampersand (
&)!
035C:4C 4B D6         69            JMP     SCRTCH          ;Ricade in NEW Applesoft
035F:                 70 * FINE DEL PROGRAMMA
```

Listato 2.

```
SOURCE    FILE #01 =>BUZ.SRC
0000:                  1 *
0000:                  2 * BUZZER
0000:                  3 * Creato
0000:                  4 * MisterAkko
0000:                  5 *
0000:                  6 * EQUate table
0000:                  7 *
0000:          C030    8 SPKR       EQU     $C030
0000:          0036    9 CSWL       EQU     $36
0000:          0037   10 CSWH       EQU     $37
0000:          FCA8   11 WAIT       EQU     $FCA8
0000:          00FF   12 LEN        EQU     $FF
----- NEXT OBJECT FILE NAME IS BUZZER
0300:          0300   13            ORG     $0300
0300:                 14 *
0300:                 15 * BUZZ
0300:                 16 * Cambiando CSW si fa passare l'output da FDED
0300:                 17 * e sostituisce i BELL con caratteri "innocui"
0300:                 18 * provvedendo da sé a suonare l'altoparlante
0300:                 19 * ma usando la routine PROBELL.
0300:                 20 *
0300:A5 36            21 BUZZ       LDA     CSWL
0302:8D 31 03         22            STA     NEWCSW
0305:A5 37            23            LDA     CSWH
0307:8D 32 03         24            STA     NEWCSW+1
030A:A9 03            25            LDA     #$03
030C:85 37            26            STA     CSWH
030E:A9 12            27            LDA     #$12
0310:85 36            28            STA     CSWL
0312:C9 07            29 ISBUZZ     CMP     #$07
0314:F0 07    031D    30            BEQ     DOBUZZ
0316:C9 87            31            CMP     #$87
0318:F0 03    031D    32            BEQ     DOBUZZ
031A:6C 31 03         33            JMP     (NEWCSW)
031D:8E 33 03         34 DOBUZZ     STX     SAVEX
0320:8C 34 03         35            STY     SAVEY
0323:20 35 03         36            JSR     PROBELL
0326:AE 33 03         37            LDX     SAVEX
0329:AC 34 03         38            LDY     SAVEY
032C:A9 96            39            LDA     #$96            ;Ctrl-V
032E:6C 31 03         40            JMP     (NEWCSW)
0331:          0002   41 NEWCSW     DS      2
0333:          0001   42 SAVEX      DS      1
0334:          0001   43 SAVEY      DS      1
0335:                 44 *
0335:                 45 * PROBELL
0335:                 46 * Il tipico "buzz" di avvertimento del programmi
0335:                 47 * in ProDOS. Nessun input.
0335:                 48 * Gotchas: distrugge tutti i registri.
0335:                 49 *
0335:A9 20            50 PROBELL    LDA     #$20
0337:85 FF            51            STA     LEN
0339:A9 02            52 PROBELL1   LDA     #$02
033B:20 A8 FC         53            JSR     WAIT
033E:8D 30 C0         54            STA     SPKR
0341:A9 24            55            LDA     #$24
0343:20 A8 FC         56            JSR     WAIT
0346:8D 30 C0         57            STA     SPKR
0349:C6 FF            58            DEC     LEN
034B:D0 EC    0339    59            BNE     PROBELL1
034D:60               60            RTS
034E:                 61 * Fine del programma


?0300 BUZZ         37 CSWH         36 CSWL        031D DOBUZZ

?0312 ISBUZZ       FF LEN        0331 NEWCSW      0339 PROBELL1

 0335 PROBELL    0333 SAVEX      0334 SAVEY       C030 SPKR

 FCA8 WAIT
** SUCCESSFUL ASSEMBLY := NO ERRORS
** ASSEMBLER CREATED ON 15-JAN-84 21:28
** TOTAL LINES ASSEMBLED     61
** FREE SPACE PAGE COUNT    89
```

ciascuna versione, oltre che di un lungo programma dimostrativo delle possibilità di Pro.dotto, potete ordinare il dischetto di Pro.dotto alla redazione, seguendo le modalità spiegate altrove su questa stessa rivista. Così facendo eviterete i noiosi problemi conseguenti all'introduzione in memoria di un programma in linguaggio macchina, dove un singolo errore potrebbe avere effetti disastrosi sul risultato finale o, peggio ancora, sui dischi e programmi sui quali verrà usato. In questa puntata tratteremo le caratteristiche generali di ProDos e di Pro.dotto, e vi verrano proposti due primi semplici e brevi programmi in linguaggio macchina, a scopo introduttivo. Anch'essi sono presenti sul dischetto succitato.

ProDos

ProDos è un sistema operativo. In quanto tale, controlla l'uso di Apple e della sua memoria, concedendola ai vari programmi che ne fanno richiesta, regolando lo scambio di informazioni e il cambio di programma in programma. Nella sua veste di sistema operativo del disco (Dos), poi, ProDos effettua tutti gli accessi al disco richiamandone programmi e dati, sotto controllo dell'utente o di un programma. Al contrario del Dos 3.3, il precedente sistema operativo dell'Apple II, ProDos è in grado di gestire tutti i disk drive disponibili per l'Apple II: il Disk II, il nuovo Unidisk da 800 Kbyte, il Profile da 5 e da 10 Mbyte. La sua massima capacità, sinora non sfruttata da nessun sistema, è di 32 Mbyte su ciascun disco in linea (cioè di 32.768 Kbyte). La velocità di trasferimento dei dati è di 8 Kbyte al secondo, circa otto-dieci volte maggiore che sotto Dos 3.3, e di gran lunga superiore a quasi tutti i personal computer disponibili sul mercato.

ProDos è costituito internamente da una serie di routine in linguaggio macchina richiamabili da un programma che ne faccia uso. Per usare ProDos, un programma deve attenersi a una serie di convenzioni fissate dalla Apple Computer nel ProDos technical reference manual; in sintesi, si tratta di chiamare ProDos come una subroutine, specificando con un numero di riferimento quale funzione si desidera vedere svolta dal sistema operativo, e aggiungendo gli eventuali parametri.

Un programma che interagisce direttamente con ProDos è chiamato programma di sistema (la terminologia è qui leggermente differente da quella usata di norma su altri computer e sui grossi computer, i mainframe). Un esempio di programma di sistema è il Basic.System, il programma che cura l'interazione tra ProDos e il Basic Applesoft, oppure il Filer. Un programma Basic, al contrario, non è un programma di sistema, perché usa il sistema operativo solo indirettamente, attraverso il Basic.System, con comandi di natura elevata (come Catalog, Load, Save). I programmi salvati sul dischetto sono differenziati a seconda della loro natura: un programma di sistema è indicato dalla sigla Sys, un programma Applesoft Basic con Bas, eccetera.

Entriamo ora più in dettaglio nel funzionamento dei dischi ProDos: queste informazioni ci serviranno per utilizzare al pieno delle sue possibilità il Pro.dotto.

Struttura di un disco ProDos

Un disco ProDos è diviso (da un punto di vista logico poiché in realtà la situazione è molto più complessa) in entità dette blocchi, costituite da 512 byte ciascuna. Un floppy disk contiene 280 di questi blocchi. Parte dei blocchi non viene utilizzata per conservare dei file, ma per custodire informazioni concernenti il disco stesso: si tratta dei blocchi numerati da zero a sette su un dischetto o sul disco Ram, sino a nove sul Profile da 5 megabyte. Notate che il ProDos stesso, al contrario di quanto avveniva con il Dos 3.3, è contenuto in un file del disco e non gode di alcun trattameto speciale da questo punto di vista.

I blocchi zero e uno contengono uno speciale programma chiamato Loader (caricatore), e vengono eseguiti al lancio del dischetto: il loro scopo è caricare dal disco il file ProDos e cedergli il comando: se questo non esiste emettono un messaggio d'errore e bloccano l'esecuzione.

I blocchi dal due al cinque compresi contengono la directory (il catalogo) del dischetto; vi sono custoditi i titoli dei file e le informazioni per recuperarli. Infine il settimo blocco (i blocchi dal sette al nove per Profile) custodisce una mappa del disco dove vengono marcati tutti i blocchi come in uso (e dunque da non scrivere, ma solo da leggere) o liberi (dunque assegnabili a nuovi file che vengano creati).

Notate che i file, le cui caratteristiche sono custodite nella directory del volume, possono essere directory a loro volta - le cosiddette sub-directory: questa struttura rende maneggevole il Profile, anche se risulta praticamente inutile su un semplice dischetto.

In ciascuno dei blocchi che contengono una directory i primi 4 byte custodiscono dei puntatori al blocco precedente e seguente che costituisce la directory. Seguono le informazioni sui file, ognuno dei quali occupa 38 byte; ma il primo spazio di 38 byte conserva le informazioni riguardanti il dischetto. I 38 byte sono così suddivisi.

Il primo byte contiene nella prima cifra uno due o tre a indicare la lunghezza del file: uno se è lungo un solo blocco, due se più di uno e sino a 256 blocchi, tre se ancora di più (praticamente mai; o almeno chi scrive non ha mai incontrato un file più lungo di 128 Kbyte). Un valore di D indica che il file è una sub-directory. La seconda cifra del prim' byte indica la lunghezza del nome del file, e i successivi 15 byte sono riservati al nome. Il byte numero 16 (in esadecimale il numero $10) indica il tipo di file; si veda la seguente tabella:

Tipo di file	Codice corrispondente
TXT	04
BIN	06
DIR	0F
USR	F1-F8
INT	FA
IVR (VAR IntBasic)	FB
BAS	FC
VAR (Applesoft)	FD
REL	FE
SYS	FF

I due byte successivi indicano il numero del primo blocco del file. Di seguito, i byte numero $13 e $14 indicano la lunghezza in blocchi del file.

I byte da $15 e $17 indicano la lunghezza del file espressa in byte, quelli da $18 a $1B la sua data di creazione nel formato che ProDos usa per data e

ora. Il byte $1C indica il numero di versione del ProDos che ha creato il file, e il successivo il numero di versione del più vecchio ProDos cui è consentito accedere al file (entrambi contengono normalmente zero).
Il byte $1E indica se il file è locked; i byte $1F e $20 contengono l'indirizzo di inizio di un programma binario o la lunghezza dei campi per un file di testo non sequenziale.

Seguono quattro byte, da $21 a $24 che contengono data e ora dell'ultima modifica al file e due byte che ripetono le informazioni dei byte $11 e $12.
Il blocco del file indicato come primo, contiene il file medesimo se questo è lungo un solo blocco, altrimenti i numeri dei blocchi che contengono effettivamente il file.
Nota: tutti i numeri espressi in due byte sono conservati nella forma tipica del microprocessore 65C02: prima il byte meno significativo, quello inferiore, poi il più significativo.

I due programmi introduttivi

Qui di seguito presentiamo due brevi programmi in linguaggio macchina. Sono listati nella forma cosiddetta sorgente, ovvero nella forma che potete introdurre in memoria servendovi di un assemblatore.
L'assemblatore usato dall'autore è l'Edasm ProDos 1.0; altri assemblatori funzionanti sotto ProDos dovrebbero poter essere usati senza difficoltà. Se non conoscete i rudimenti del linguaggio macchina, tuttavia, difficilmente possederete un assemblatore o saprete come usarlo; d'altra parte, potrebbe esservi utile conservare una copia del sorgente solo se, conoscendo i meandri del linguaggio macchina, potreste essere in grado di apportare modifiche al programma, in modo da adattarlo meglio alle vostre esigenze.

Se non vi interessate al linguaggio macchina, è sufficiente introdurre il programma nella sua forma compresa e usata dall'Apple, la forma cosiddetta assemblata, dove ai simboli Assembler sono stati sostituiti i codici numerici che il microprocessore dell'Apple comprende ed esegue. In questo caso, dopo aver lanciato un dischetto con ProDos che contenga sufficiente spazio disponibile per il programma, dovrete seguire il seguente procedimento:

- entrate in Sistem Monitor battendo Call–151 <Return>;
- battete l'indirizzo di partenza del programma: potete scoprirlo osservando nel listato sorgente il numero (esadecimale) che segue la riga -- Next object file name is... Nel caso di entrambi i programmi che vi proponiamo qui, questo indirizzo vale 300;
- battete di seguito ":" (due punti);
- batttete i codici numerici esadecimali che leggete nella prima colonna del listato. Nel caso del programma FP, dunque,dovreste battere:

300: 20 F8 BE A9 00 8D 00 08 8D 01 08

eccetera. Ogni dieci-venti codici, battete Return e riprendete battendo "due punti" e poi i codici successivi;

- al termine tornate in Basic battendo Control-y <Return>;
- salvate il programma battendo il comando Bsave (nome del programma), A$ (indirizzo iniziale). E$ (indirizzo finale)

Troverete l'indirizzo finale come ultimo indirizzo nel listato sorgente. Nel caso del programma FP questo comando sarebbe dunque Bsave Fp, A$300, E$35F.
FP è un programma in linguaggio macchina che esegue le incombenze di un comando disponibile sotto Dos 3.3 ed eliminato in ProDos Basic.System, è tuttavia di grande utilità. È una specie di super New che risistema le locazioni interne della macchina, controlla che Applesoft non si inceppi, e che provvede anche a eliminare ogni programma che esiste in memoria insieme a Basic.System: può dunque venire utilizzato per rimuovere Help e/o Editor di blocchi della memoria. Nel programma Help di Pro.dotto esiste una funzione simile, che tratteremo nella puntata successiva. Buzzer è un semplice programma, pure in linguaggio macchina, che modifica il tipico "bip" dell'altoparlante di Apple II con un "buzz" più morbido e meno spaccatimpani, il tipico suono dei programmi ProDos, usato da tutti i componenti di Pro.dotto come da Filer, Convert e altri programmi di sistema ProDos. Reset lo annulla e Call 768 lo reinnesca. ■

(Continua)

M20 Paint

Disegnare facile con l'Olivetti

di Sergio Borsani

Il nome è ambizioso: ricorda un po' troppo il famoso programma creato per il Mac della Casa di Cupertino. Un confronto non si può nemmeno azzardare, tuttavia basta uno sguardo ai disegni e ai particolari testi elaborati dal computer, per comprendere come M20 Paint, pur nella sua semplicità, possa svolgere funzioni non trascurabili. I programmi rivolti al Cad (Computer aided design) godono di un'eco sempre più vasta tra il grande pubblico; purtroppo restano ancora poco accessibili poiché l'elaborazione grafica richiede una grande memoria centrale e di massa, velocità di esecuzione e spesso necessita di schede di espansione del sistema con un ovvio aumento dei costi. I programmi per così dire casalinghi, si limitano in genere alle funzioni più semplici cadendo spesso in difetto per ingenuità. Se ci si limita a mettere in On i pixel del monitor sotto il controllo della tastiera per poi stampare l'immagine così ottenuta, la battaglia è persa. Qualsiasi profano non esiterà ad ammettere che un mezzo tradizionale come il "rapidograph" è più efficiente e dà una resa migliore. Per essere competitivo il computer deve poter sfruttare le sue capacità di elaborazione e permettere il raggiungimento di risultati, in altro modo difficilmente conseguibili.

Tralasciando alcune caratteristiche peculiari dei programmi professionali dedicati, per esempio a progettisti e architetti, ciò che si può chiedere, oltre agli indispensabili comandi per creare l'immagine sul video, è la presenza di funzioni rivolte alla memorizzazione dei disegni e al loro richiamo, la possibilità di separare un'immagine negli elementi che la compongono e, al contrario, la possibilità di operare un collage non solo tra elementi presenti sullo schermo, ma anche tra questi e file grafici contenuti su disco. Alcuni programmi consentono la copertura delle superfici con svariati retini come il tratteggio. È auspicabile, riteniamo, anche la gestione contemporanea della grafica e del testo.

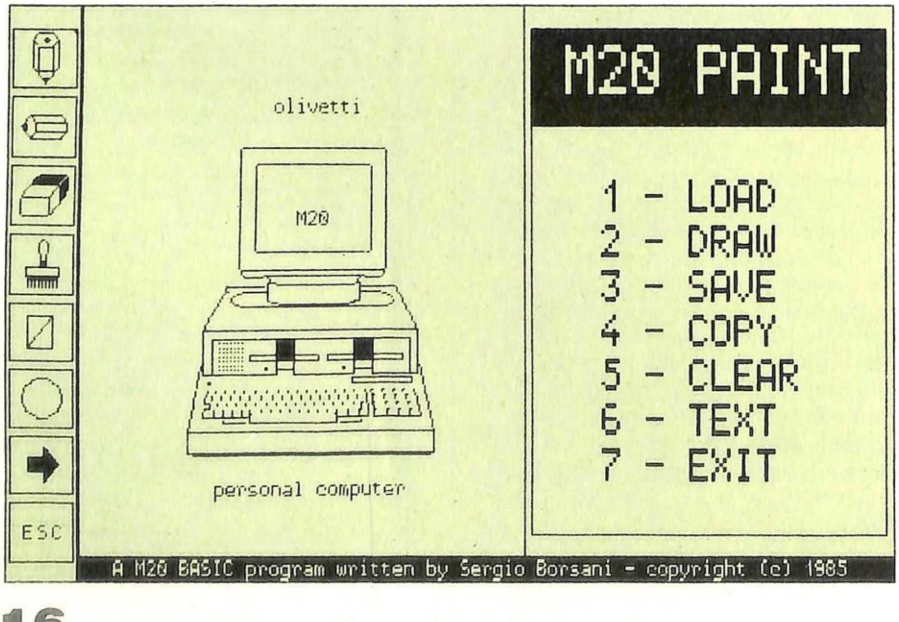


Figura 1 - Ambiente di lavoro di M20 Paint. Alla destra il menu principale, a sinistra quello associato all'esecuzione grafica, in basso una riga per messaggi e opzioni di stampa, al centro la finestra grafica. Il menu principale si sovrappone come un foglio senza alterare o limitare la parte grafica.

M20 Paint, a parte la capacità di creare particolari retini, possiede tutte le caratteristiche su indicate. Uno sguardo alla figura 1 chiarisce l'ambiente in cui si opera. Alla destra il menu principale che si sovrappone come un foglio alla finestra grafica senza alterarne il contenuto. Sulla sinistra i simboli grafici per selezionare le varie funzioni operative. In basso una riga riservata all'introduzione dei dati e ai messaggi.

(1) Load: richiama in memoria un file di dati precedentemente creato con M20 Paint; il file contiene un'immagine (sotto forma di numeri) di dimensioni variabili che viene riprodotta sullo schermo anche durante l'esecuzione di un altro disegno.

(2) Draw: fa apparire il cursore grafico (un puntino) che accompagna e guida la realizzazione delle immagini sotto il controllo dei tasti unitamente al menu con i simboli grafici posto sul lato sinistro del video.

(3) Save: memorizza l'intero schermo o una sua parte con un nome specificato dall'utente; a esempio, 1:robot. Lo stesso nome verrà specificato con il comando Load del menu.

(4) Copy: esegue l'hard copy tramite la stampante (PR 1450). In questa versione del programma viene copiata l'intera finestra grafica in bit image e non è possibile diminuire la superficie da riprodurre. In ogni modo la stampa comporta solo una manciata di secondi.

(5) Clear: pulisce lo schermo o una sua parte, una superficie rettangolare della quale si devono specificare due punti diametralmente opposti. Se non si vuole perdere irrimediabilmente il disegno, esso va salvato prima di usare questo comando.

(6) Text: abilita il modo testo; si ha così a

disposizione un diverso cursore che si sposta in ogni posizione della finestra grafica per l'inserimento di caratteri alfanumerici. In una prossima versione il programma disporrà di un secondo set di caratteri, personalizzato, con simboli doppi del normale per intestazioni, titoli, eccetera. Premendo il tasto Cr si esce dal modo testo e si torna al menu principale.

(7) Exit: si esce dal programma.

I simboli grafici presenti sulla parte sinistra dello schermo hanno un chiaro significato; a essi si accede dopo aver selezionato l'opzione (2) Draw. A questo punto scompare il menu principale e ricompaiono eventuali disegni ai quali si era sovrapposto. Le funzioni grafiche si selezionano tramite i tasti con le frecce (8) e (2) e si confermano con Cr o il tasto Enter che dir si voglia. Il simbolo prescelto e reso attivo appare in reverse, cioè a colori invertiti. Dall'alto verso il basso abbiamo: pen down (il cursore scrive), pen up (il cursore si sposta senza scrivere, la gomma (il cursore cancella), il pennello (si riempie una superficie delimitata a partire dalla posizione del cursore), la linea e il rettangolo (viene tracciata una linea o un rettangolo), il cerchio (viene tracciato un cerchio o un'ellisse), la freccia (viene spostato o copiato un disegno presente sul video), escape (si torna al menu principale).

Dopo aver selezionato (2) Draw è attiva la funzione pen up. Il cursore grafico possiede, per così dire, due marce per consentire il suo trasferimento a diverse velocità. Premendo il tasto (+) si sposta di dieci pixel alla volta, con il tasto (–) il movimento avviene un pixel alla volta. Per attivare il simbolo pen down si usa la freccia verso l'alto, (Shift) (8); premendo il tasto (5) = Home, si torna da ogni altro simbolo a pen up. È anche possibile fissare un mark-point per il cursore con il tasto del punto (.); memorizzata così una posizione si può tornare immediatamente a essa con (Shift) (5) = Home (questa volta premendo anche il tasto delle maiuscole). Durante l'esecuzione di un disegno si dispone solo della minore velocità di trasferimento e per lunghi tratti rettilinei è preferibile utilizzare la funzione Line che vedremo tra poco. Il movimento del cursore avviene in otto direzioni, le quattro principali con i tasti muniti di frecce e secondo le bisettrici usando i tasti intermedi. Proseguendo, la gomma mette in Off i punti e funziona come pen down a colori invertiti; con essa, a esempio, si può scrivere in nero su uno sfondo bianco. Il pennello riempie una superficie a partire dal punto in cui si trova il cursore. È bene ricordare che l'istruzione Paint necessita di una superficie delimitata da una qualsiasi curva chiusa; se esiste anche la più piccola apertura, il colore passa e riempie l'intera finestra. Il pennello si può usare ripetutamente senza tornare a pen up. Il simbolo a forma di rettangolo con la sua diagonale si usa per tracciare linee e rettangoli. Dopo averlo selezionato con Cr si possono specificare alcuni parametri dei quali vengono proposti i valori di default; se essi non devono essere modificati è sufficiente confermarli con Cr. Gli estremi del segmento o del rettangolo si specificano spostando il cursore e confermando con il tasto Cr. Interessante è la presenza delle funzioni logiche o booleane; esse hanno lo stesso significato riportato nel manuale del Basic. In particolare con Or si sovrappone un'immagine a una preesistente senza cancellarla; con Xor si sovrappongono ugualmente due immagini, ma i pixel che formano la seconda vengono messi in On se sotto trovano pixel in Off e in Off se trovano pixel in On; in tal modo i due disegni, pur sovrapposti, mantengono la loro individualità.

Per il cerchio valgono regole analoghe e i parametri qui usati sono ben descritti nel manuale. Aspect ratio determina la forma: cerchio o ellisse; il valore di default si riferisce a un cerchio. I due punti da indicare qui, con il cursore, rappresentano il centro il primo, e un punto sulla circonferenza il secondo.

Per non perdere di vista il centro durante lo spostamento del cursore è opportuno segnare il punto con pen down e poi eventualmente cancellarlo.

Il simbolo con la freccia viene usato per spostare o copiare un disegno da una parte all'altra dello schermo. Se il disegno viene spostato si cancella automaticamente l'originale, altrimenti resta e si crea una copia (vedi figure 2 e 3).

In ogni caso la superficie è di forma rettangolare e si delimita ponendo successivamente il cursore su due vertici diametralmente opposti. La nuova posizione si specifica invariabilmente usando il vertice in alto a sinistra. Con questa funzione, naturalmente, si spostano anche caratteri alfanumerici e righe di testo per meglio adattarli a grafici e diagrammi.

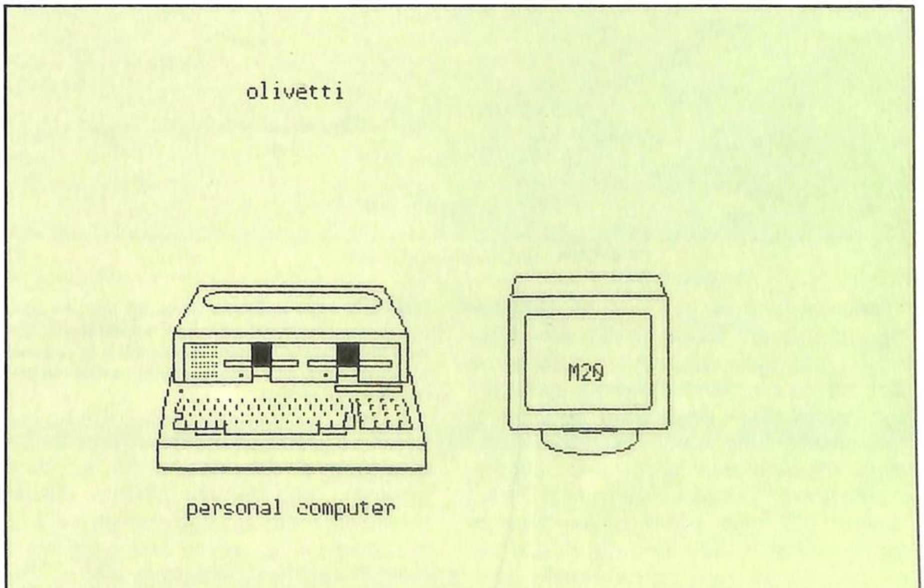


Figura 2 - La funzione "sposta" permette la scomposizione di un disegno nei suoi elementi. Essa risulta utile anche quando si voglia posizionare una scritta all'interno di un disegno senza essere condizionati dalle righe e dalle colonne di stampa.

Le figure pubblicate sono indicative. Il programma si presta a diverse applicazioni dove è presente una certa ripetitività o dove un testo è intimamente legato alla grafica e richiede l'uso di simboli altrimenti non disponibili come accade con relazioni e testi scientifici. La figura 4 mostra delle formule di fisica e un grafico che illustra la differenza di due vettori. I caratteri in grassetto sono stati creati una prima volta e poi copiati con l'apposita funzione: anche gli altri caratteri alfanumerici che appaiono nel grafico sono stati scritti altrove nel modo testo e poi posizionati all'interno del piano cartesiano tra le linee che rappresentano i vettori. La figura 5 è un altro esempio di fusione tra grafica e testo; l'argomento riguarda la geometria. Nessun word processor per personal computer potrebbe soddisfare le stesse esigenze.

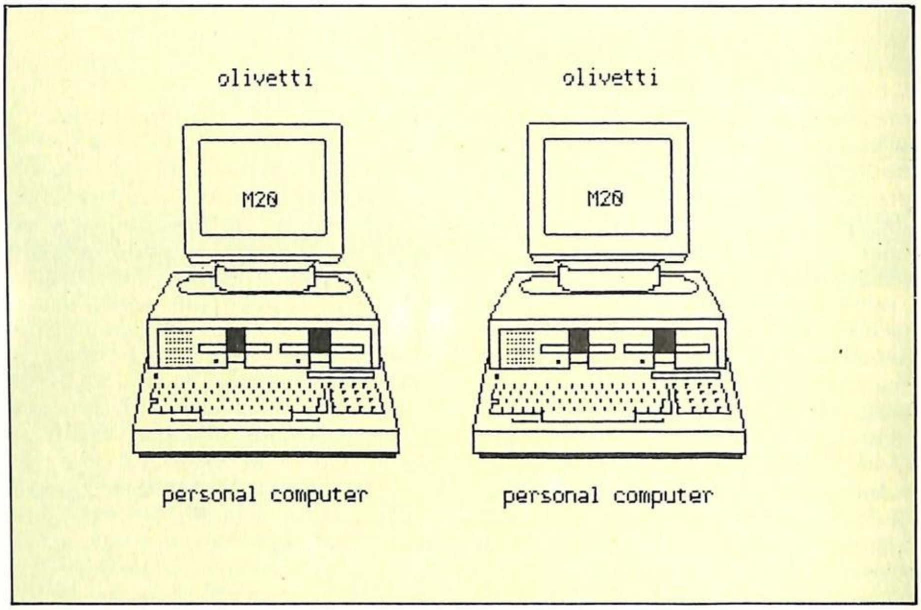


Figura 3 - Associato a "sposta" troviamo il comando copia, anch'esso richiamabile attivando il simbolo a forma di freccia. Un'importante applicazione si ha quando si usano le funzioni sposta/copia con caratteri e simboli personalizzati richiamabili in ogni momento da disco.

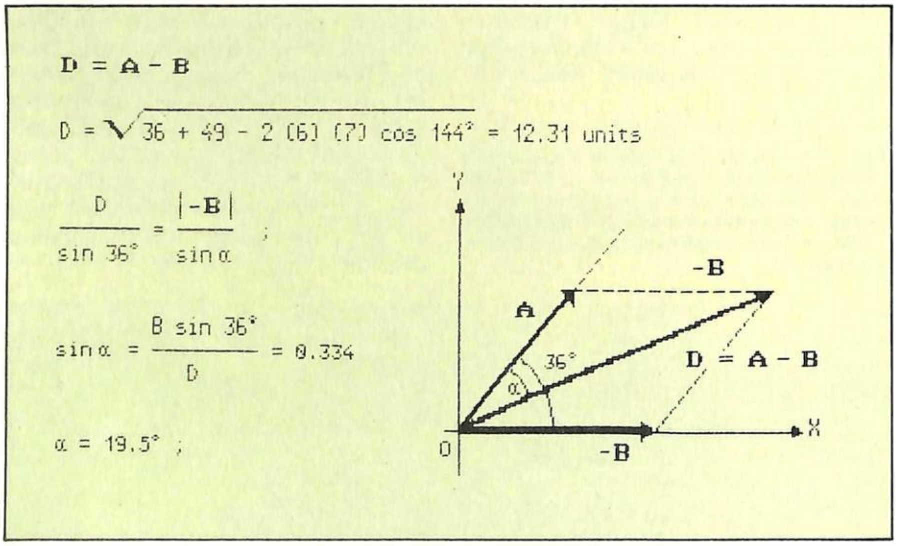


Figura 4 - Un esempio di come sia possibile integrare i normali caratteri alfanumerici con caratteri speciali anche ingranditi o in grassetto. L'esigenza è particolarmente sentita nei testi tecnici e scientifici.

Nelle altre figure (6 e 7) predomina un tipo di grafica con elementi a volte ripetitivi. L'uso delle superfici chiare e scure contribuisce a dare risalto alle immagini e a evidenziare particolari strutture. Il disegno che solitamente viene chiamato a "mano libera" e che per trasposizione potremmo qui chiamare a "tasto libero" (?) non trova evidentemente molto spazio e, alla luce delle considerazioni iniziali, non ha nemmeno sufficienti ragioni di esistere. Forse con una tavoletta grafica... Concludiamo con il triangolo di Penrose, detto anche il triangolo impossibile, portato spesso come esempio di illusione ottica e di inganno per la mente che si rifiuta di percepire delle linee che rievocano una disposizione spaziale degli elementi (i lati del triangolo) che non può esistere nella realtà (vedi figura 7). Sarà impossibile per la mente, ma non certo per M20 Paint.

Un particolare tecnico

Il programma utilizza principalmente le istruzioni grafiche Pset, Line, Circle, Paint, eccetera, le quali sono ben illustrate nel manuale del Basic e sono trasparenti all'utente. Forse è meno noto il modo in cui l'M20 memorizza le immagini. A tale scopo esiste l'istruzione Get che trasferisce in un vettore numerico una sequenza di punti appartenenti a una regione rettangolare dello schermo. Il sistema fa corrispondere un valore intero, compreso tra –32.768 e +32.767, a ogni sequenza di 16 punti. Nel vettore vengono anche inserite informazioni sulle dimensioni della superficie da memorizzare e sul colore.

Tutto ciò che deve fare il programmatore è dimensionare una variabile con

NOTE

Concorso 7BIT

IMPARARE LA MUSICA COL C64

...Vinci 30 Commodore Plus/4

Ecco l'elenco dei 30 fortunati vincitori

D.M. 4/275256 DEL 2/3/1985

1) **TRENTI GABRIELE -**
Via Delle Tovaglie, 14 - 40124 BOLOGNA
2) **LA SCALEA TONINO -**
Via Figurella, 26 - 87011 CASSANO ALL'IONIO (CS)
3) **PASQUETTO PAOLO -**
P.zza Europa, 5 - 10059 SUSA (TO)
4) **DE PALO CATALDO -**
Via Angiulli, 10 - 70037 RUVO DI PUGLIA (BA)
5) **SAGGIOMO GIOVANNI -**
Via Cimarosa, 180 - 80127 NAPOLI
6) **DECCIO STEFANO -**
V.le Angelico, 173 - 00195 ROMA
7) **NAPOLANO GIACOMO c/o Monaco Silvana -**
Via Menaggio - 22010 SAN PIETRO SOVERA (CO)
8) **FRANCINI IVAN -** Via Castellonchio, 49
50067 RIGNANO SULL'ARNO (FI)
9) **STEFANELLI ANTONIO -** Via Manzoni, 253
80046 S. GIORGIO A CREMANO (NA)
10) **CRAVANZOLA CRISTIANO -** Via Prof. Oliva, 16
12011 BORGO S. DALMAZZO (CN)
11) **RICCI VALERIO -**
Via Centrale, 94 - 06013 LAMA (PG)
12) **MANTOVANI GIOVANNI -**
Via Repubblica, 57/bis - 27049 STRADELLA (PV)
13) **BORDIGNON LIVIO -**
Via Pio X, 42 - 31037 LORIA (TV)
14) **URSO DAVIDE c/o Bigatti -**
Via E. Castellotti, 15 - 20075 LODI (MI)
15) **BALDI F. LUCIANO -**
Corso Sebastopoli, 198 - 10136 TORINO
16) **BASSANO CARLO -**
Casella Postale, 200 - 20020 ARESE (MI)
17) **MORAN RODOLFO -**
Via Chiadino, 67 - 34142 TRIESTE
18) **CROCIANI GINO -** Via Poggio Fiorito, 16
00046 GROTTAFERRATA (Roma)
19) **DE DONNO DONATO -**
Via Canova, 146 - 73042 CASARANO (LE)
20) **GRASSI SABRINA -**
Via Torre, 127 - 72022 LATIANO (BR)
21) **ANZIVINO ANTONIO -**
Via Conti Montecchi, 8 - 66054 VASTO (CH)
22) **OLIVIERI DINO -**
Via Tofane, 2 - 10141 TORINO
23) **DE PAOLIS SERGIO -**
Via Lago D'Iseo, 17 - 73013 GALATINA (LE)
24) **IACOLETTI LUCIANO -**
Via Ribera, 5 - 80128 NAPOLI
25) **GARGANI PAOLO -**
Via A. Corelli, 58 - 50127 FIRENZE
26) **LOPARCO ROBERTO -**
Via Della Stazione di Settebagni, 19 - 00138 ROMA
27) **ZONTA FRANCESCO**
Via Carmini, 16 - 36078 VALDAGNO (VI)
28) **DI NARDO ENRICO -**
Via S. Esposito, 10 - 83100 AVELLINO
29) **TALLARITA Ing. PIETRO -**
Via Del Pesco, 13 - 90147 PALERMO
30) **GILLI FRANCESCO -**
Via Redipuglia, 75 - 09100 CAGLIARI

Congratulazioni!

GRUPPO EDITORIALE JACKSON
DIVISIONE GRANDI OPERE

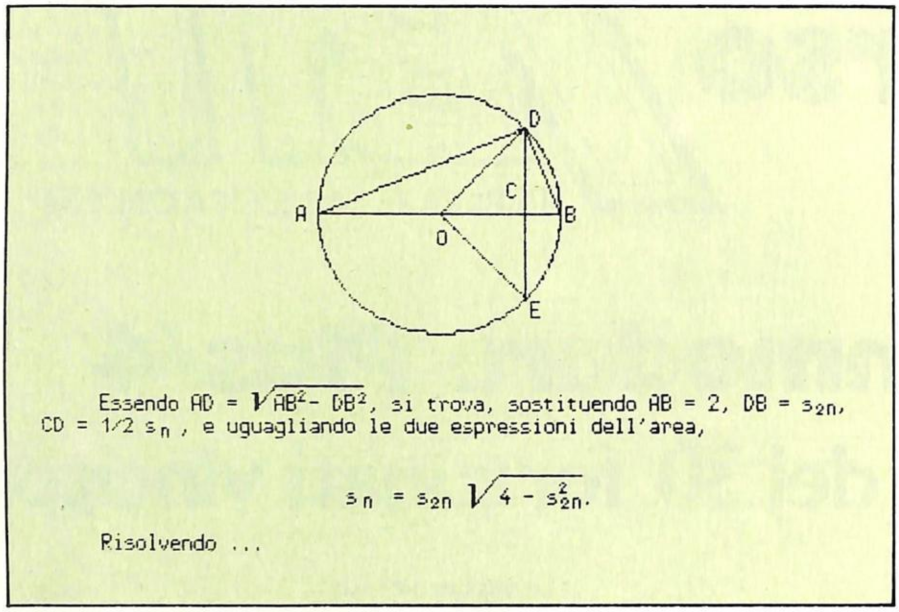


Figura 5 - Un ulteriore esempio di fusione di testo e grafica. Il comando Pcos Sp che riproduce su carta lo schermo in bit image, provoca una leggera distorsione sui caratteri e lievi irregolarità che tuttavia non disturbano la lettura.

indice in modo che gli elementi siano sufficienti a contenere tutti i valori numerici relativi all'immagine, a esempio, Dim A% (1000). In secondo luogo, creato un disegno, userà l'istruzione Get specificando la superficie da memorizzare. Get (200,100)–(231,131),A%(0), si riferisce a una porzione di 32 x 32 punti; i valori tra parentesi sono le coordinate di due vertici diametralmente opposti che racchiudono il disegno.

È utile sapere che si può generare un'immagine in memoria e poi restituirla con l'istruzione Put anche senza realizzarla materialmente sullo schermo, ma caricando opportuni valori interi nel vettore a%(). Così sono stati realizzati i simboli raffiguranti le matite, la gomma, il pennello, eccetera; essi sono generati dalle sequenze numeriche contenute nelle istruzioni Data a partire dalla linea 1850 fino alla fine del listato.

Qual è il segreto di quei numeri? Che cosa significano? Sarà lo stesso computer a risolvere la crittografia. Provate a eseguire un disegno usando le istruzioni Pset, Line, eccetera e a memorizzarlo con una Get. Al termine indagate sul contenuto del vettore a%(). For J = 0 To <n> : Print A%(J); : Next J. I primi due numeri sono rispettivamente la larghezza e l'altezza del rettangolo che delimita la superficie, il terzo è un codice di colore, 256 per uno schermo in bianco e nero; tutti i numeri che seguono si riferiscono alla disposizione dei punti. Come si è già detto, il computer ne considera 16 alla volta; ogni punto in posizione Off, cioè spento, ha valore zero, quelli nello stato On hanno un valore che dipende dalle potenze di due. Chi possiede una certa pratica di numerazione binaria avrà già compreso come funziona il procedimento. Il primo punto a destra (dei 16) vale uno; gli altri, procedendo verso sinistra, valgono due, quattro, otto, 16, 32 e così via fino a 32.768. Il numero intero da associare all'intera sequenza è dato dalla somma di tutti i valori dei punti in On. Il problema è che esso non può essere maggiore di 32.767. Quando ciò si verifica il sistema esegue il cosiddetto "completamento a due" del numero binario; nell'ambito del sistema decimale si dovrà procedere a calcolare la differenza tra il numero ottenuto e 65.536 ricavano così un intero negativo non inferiore a –32768. A esempio, se tutti i 16 punti dovessero essere in On la somma dei loro valori sarebbe 65.535; si calcola allora 65535–65.536 = –1 e questo è il numero da inserire nel vettore a%() per esprimere tale configurazione di punti.

Il metodo va usato con una certa cautela. Dei valori errati possono causare un blocco del sistema rendendo necessario un Reset per poterlo riavviare.

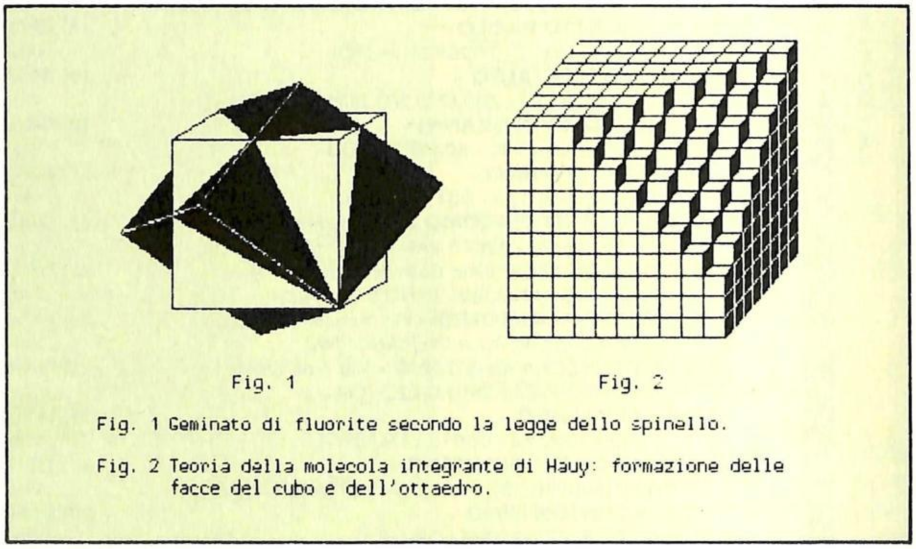


Figura 6 - Effetti prospettici e l'uso dell'ombreggiatura possono creare belle immagini tridimensionali.

Descrizione del listato

90- Si rendono residenti alcuni comandi del Pcos.

100-120 - Si danno i valori iniziali ad alcune variabili e si suddivide lo schermo in tre finestre.

130-170 - Vengono caricati i vettori che rappresentano i simboli grafici del menu di sinistra.

Tali simboli vengono poi visualizzati con l'istruzione Put.

180-250 - Si crea il menu principale sulla destra sovrapponendolo alla finestra grafica.

260-300 - Le linee accettano la scelta dell'utente e tolgono il menu ripristinando la finestra grafica.

310-430 - Viene eseguito il comando Load. La linea 320 apre il file, mentre la lettura viene effettuata alla linea 340. Il numero di record è sempre un multiplo di dieci.

440 - Inizia la routine per il comando Draw.

500-650 - Routine per accettare le istruzioni dalla tastiera.

660-720 - Individuazione della funzione prescelta.

730-870 - Esecuzione di un'istruzione Line/Box. Le numerose linee sono giustificate dal fatto che i parametri presenti nell'istruzione Line non possono essere specificati tramite una variabile di stringa.

880-990 - Esecuzione di un'istruzione Circle. La linea 890 provoca volutamente un errore rinviando il programma a una linea inesistente, in modo da trasferire il controllo alla subroutine 1400 legata all'istruzione On Error Goto.

1000-1100 - Esecuzione del comando sposta/copia.

1100-1180 - La routine memorizza tutto lo schermo, o una sua parte, in un file residente su disco.

1190-1250 - Esecuzione dell'hardcopy di ciò che appare sul video in un dato istante. Per centrare l'immagine sulla carta vengono chiuse le finestre due e tre: si sfrutta così lo spazio vuoto che si crea sulla parte sinistra del video.

1260-1300 - Routine per cancellare la finestra grafica o una sua parte.

1310-1390 - Si accettano caratteri alfanumerici. Le funzioni di editing sono ridotte all'essenziale. Sono abilitati solo i quattro tasti per il movimento del cursore e il tasto Cr, oltre naturalmente a tutti gli altri per la scrittura.

1400-1410 - Subroutine per il controllo degli errori.

1420 - Controllo della tastiera.

1430 - Cancellazione dei messaggi che appaiono nell'ultima riga in basso.

1440 - Scrittura in reverse di un'opzione attiva.

1450-1570 - Subroutine per accettare il nome di un file. Allo scopo non si è usata una semplice istruzione Input per poter limitare la lunghezza della stringa a non più di 14 caratteri.

1580-1700 - Sottoprogramma di supporto a varie funzioni e comandi. Con esso si accettano le coordinate di due punti dello schermo.

1710-1780 - Sottomenu per la scelta degli operatori booleani.

1790-1840 - Subroutine per la scelta delle opzioni sposta e copia.

1850-2380 - Istruzioni Data contenenti i valori numerici che definiscono la forma dei simboli presenti nel menu di sinistra, cioè la matita, la gomma, il pennello, eccetera.

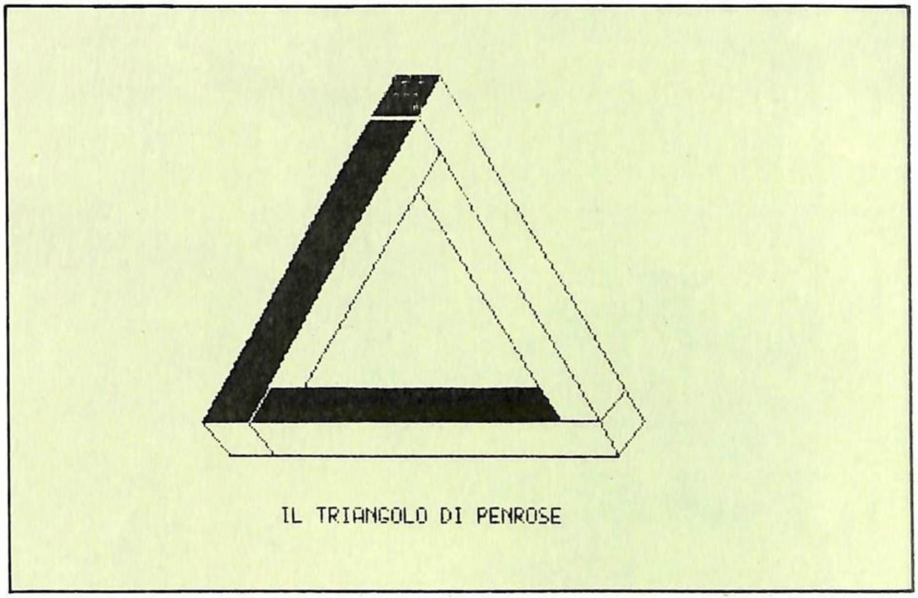


Figura 7 - Il triangolo di Penrose, detto anche il triangolo impossibile per il diverso orientamento prospettico dei suoi vertici. In fase di stampa si può scegliere se lasciare la figura incorniciata oppure no.

Principali variabili del programma

s%() - Memorizza il contenuto della finestra grafica o di una sua parte in bit image.

a%() - Memorizza di volta in volta un simbolo visibile nel menu alla sinistra.

bo$() - Operatori booleani. And, Or, Xor, eccetera.

xmark, ymark - Coordinate del mark point.

x, y - Coordinate del cursore.

r - Velocità di trasferimento del cursore. il suo valore può essere uno o dieci.

yp - Indica l'ordinata del simbolo attivo del menu di sinistra.

m$ - Contiene di volta in volta un diverso messaggio che apparirà nella finestra numero tre, la riga in basso.

xk - Posizione iniziale della stringa da trasferire alla subroutine con funzione di data entry.

1$ - Stringa da trasferire alla subroutine con funzione di data entry.

modo% - Contiene il numero progressivo della funzione attiva durante l'esecuzione di un disegno.

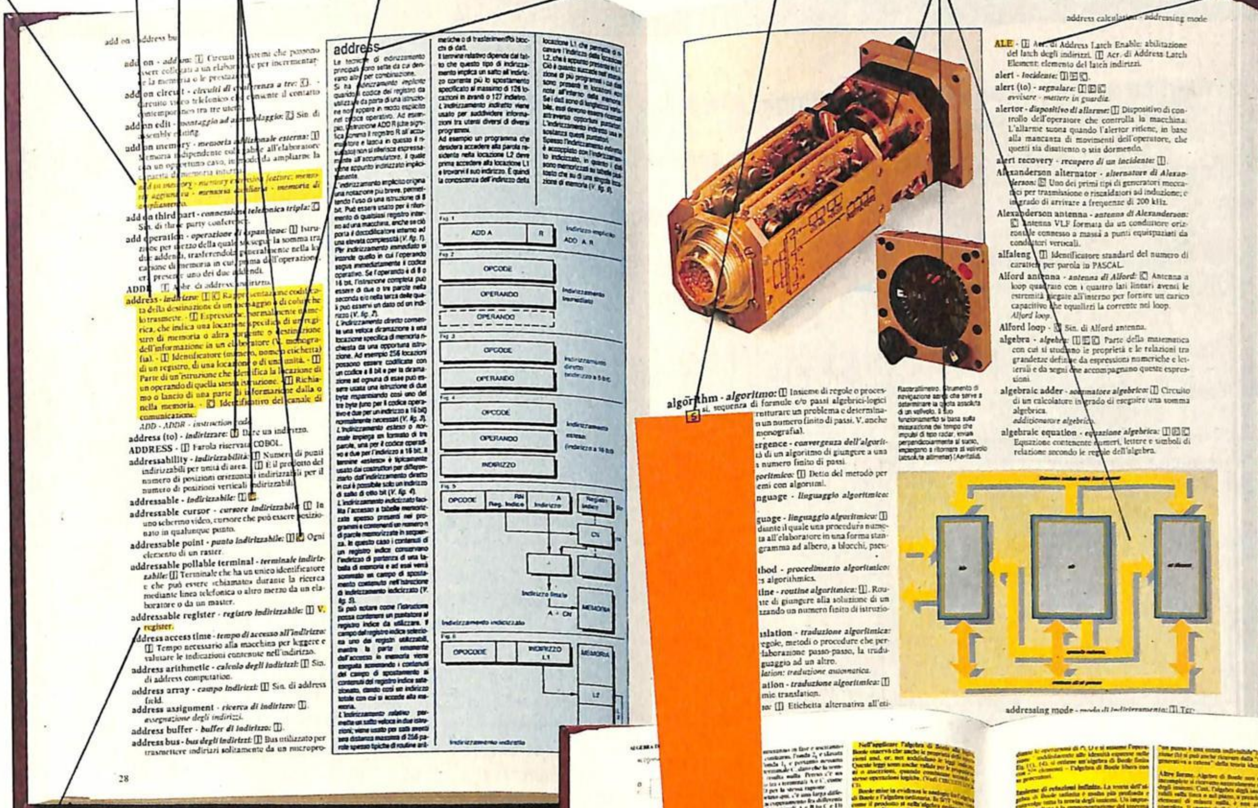
dx,dy - Coordinate del cursore grafico relative all'ultima posizione occupata.

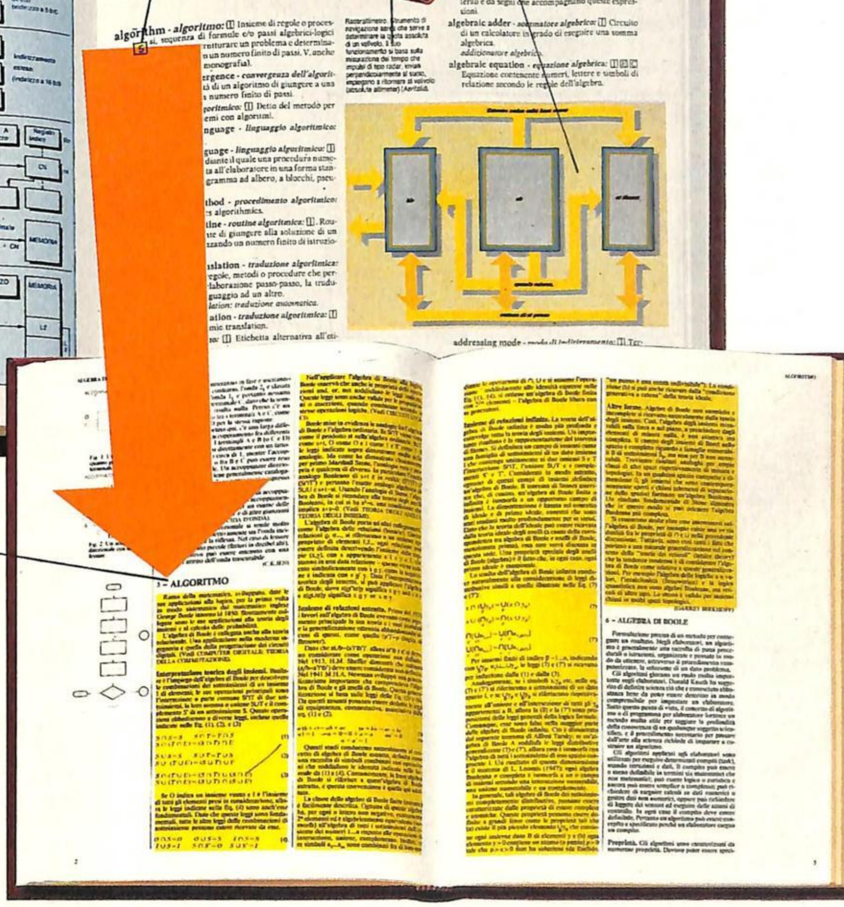
e - Eccentricità di un'ellisse.

rd - Raggio di un cerchio o semiasse maggiore di un'ellisse.

elemat - Numero di elementi che com-


IL GRANDE
DIZIONARIO
JACKSON
DI ELETTRONICA
E INFORMATICA
LA SCIENZA DI OGGI
LA CULTURA DI DOMANI.
IL GRANDE DIZIONARIO JACKSON DI ELETTRONICA E INFORMATICA
MONOGRAFIE
Philips Computers
10 Volumi IL GRANDE DIZIONARIO JACKSON DI ELETTRONICA E INFORMATICA
2 Volumi MONOGRAFIE





Il Grande Dizionario Jackson di Elettronica e Informatica è il risultato di un importante lavoro di ricerca, durata più di cinque anni, che ha coinvolto decine di specialisti.

DEI è l'unica opera al mondo che risponde all'ormai improrogabile necessità di raccogliere e organizzare tutti i termini tecnici nati dalla diffusione massiccia dell'elettronica, informatica e comunicazione per formalizzare e consolidare il "nuovo sapere".

DEI è per tutti coloro che hanno capito l'importanza di sapersi orientare in questo nuovo universo linguistico; per chi vuole comprendere il significato di termini che le nuove discipline hanno introdotto anche nel linguaggio comune; per quanti desiderano saperli utilizzare con padronanza per non rimanere esclusi da un mondo che sta compiendo un cambiamento le cui dimensioni sono senza precedenti; per coloro, infine, che per lavoro, studio, informazione vogliono possedere, già da oggi, la "cultura di domani".

Con DEI, infatti, la trasformazione della scienza di oggi nella cultura di domani compie un decisivo passo avanti.

DEI
- **45000 termini**
- **3000 illustrazioni**
- **1000 monografie in 100 fascicoli settimanali da rilegare in 12 splendidi volumi.**

IN EDICOLA IL 1° E IL 2° FASCICOLO A SOLE L. 2.500



GRUPPO EDITORIALE JACKSON
DIVISIONE GRANDI OPERE

IN COLLABORAZIONE CON PHILIPS

pongono la matrice unidimensionale s%() e che definiscono la bit image di un disegno.
xt, yt - Coordinate del cursore di testo.
k$ - Contiene il valore di un tasto premuto.
xx - Posizione relativa del cursore durante la scrittura per l'accettazione di una stringa, in questo caso il nome di un file.
ls - Lunghezza della stringa da accettare.
xl, yl - Coordinate di un punto sullo schermo.
x2, y2 - Coordinate di un secondo punto sullo schermo. Vengono usate con le istruzioni Line, Circle e Get.
xp - Indica l'inizio di ogni parola che forma il sottomenu degli operatori logici. L'opzione attiva viene stampata in reverse e la variabile xp compare nell'istruzione Cursor. ■

Listato 1 - Il programma M20 Paint.

```
10 REM  *********************************
*************
20 REM  *
         *
30 REM  *                   M20 PAINT
         *
40 REM  *
         *
50 REM  *    a program written by Sergio
 Borsani     *
60 REM  *          copyright (C) 1985
         *
70 REM  *
         *
80 REM  *********************************
*************
90 CALL "pl sf,la,sp" : CALL "sf ,,0" :
CALL "sf on"
100 CLOSE WINDOW : SCALE 0,511,0,255 : D
IM S%(8460),A%(90) : CLS : XMARK=220 : Y
MARK=120 : X=220 : Y=120 : R=1
110 BO$(1)="AND" : BO$(2)="XOR" : BO$(3)
="OR" : BO$(4)="NOT" : BO$(5)="PSET" : B
O$(6)="PRESET"
120 W1=WINDOW(0,0,10,6) : W2=WINDOW(2,7)
 : W3=WINDOW(1,245)
130 FOR J=1 TO 3 : WINDOW %J : LINE(0,0)
-(511,255),,B : NEXT J
140 WINDOW %2 : FOR J=1 TO 8
150 FOR K=0 TO 89 : READ A%(K) : NEXT K
160 PUT(10,276-J*30),A%(0) : NEXT J
170 GET(10,216)-(500,188),A%(0) : YP=216
180 WINDOW %1 : SCALE 0,439,0,244 : GET(
0,244)-(439,0),S%(0)
190 W4=WINDOW(3,40) : WINDOW %4 : LINE(0
,0)-(511,255),,B : LINE(10,10)-(500,245)
,,B
200 LINE(10,245)-(500,200),,BF : COLOR 0
,1 : EXEC "la 'M20 PAINT',20,200,3,0,0"
: COLOR 1,0
210 EXEC "la '1 - LOAD',40,150,2" : EXEC
 "la '2 - DRAW',40,130,2"
220 EXEC "la '3 - SAVE',40,110,2" : EXEC
 "la '4 - COPY',40,90,2"
230 EXEC "la '5 - CLEAR',40,70,2" : EXEC
 "la '6 - TEXT',40,50,2"
240 EXEC "la '7 - EXIT',40,30,2" : ON ER
ROR GOTO 1400
250 M$="  M20 PAINT  (version 2.0) -  by
 Sergio Borsani - copyright (c) 1985 " :
 GOSUB 1440 : CURSOR(1,1)0
260 GOSUB 1420
270 N=ASC(K$) : IF N<49 OR N>55 THEN 260
280 IF N=55 THEN CLOSE WINDOW : CLS : Q=
WINDOW(0,0,16,8) : SCALE 0,511,0,255 : C
URSOR POINT(0,0)0 : END
290 CLOSE WINDOW %4 : WINDOW %1 : SCALE
0,439,0,244 : PUT(0,244),S%(0)
300 ON N-48 GOTO 310,440,1110,1190,1260,
1310
310 M$="File name? " : GOSUB 1440 : XK=1
4 : L$=SPACE$(14) : GOSUB 1450
320 OPEN "I",#1,L$ : C=0
330 IF EOF(1) THEN CLOSE #1 : GOTO 350
340 FOR J=C TO C+9 : INPUT #1,S%(J) : NE
XT J : C=C+10 : GOTO 330
350 M$="Indica la posizione con il curso
r point. " : GOSUB 1440 : FLAG=0: WINDOW
 %1 : GOSUB 1590
360 GOSUB 1710 : WINDOW %1 : ON ACTV GOT
O 370,380,390,400,410,420
370 PUT(X2,Y2),S%(0),AND : GOTO 430
380 PUT(X2,Y2),S%(0),XOR : GOTO 430
390 PUT(X2,Y2),S%(0),OR : GOTO 430
400 PUT(X2,Y2),S%(0),NOT : GOTO 430
410 PUT(X2,Y2),S%(0),PSET : GOTO 430
420 PUT(X2,Y2),S%(0),PRESET
430 LINE(0,0)-(439,244),,B : GOTO 440
440 WINDOW %1 : SCALE 0,439,0,244 : MODO
%=2 : WINDOW %2 : GOTO 460
450 WINDOW %2 : PUT(10,YP),A%(0) : IF MO
DO%=1 OR MODO%=3 THEN R=1
460 YP=276-MODO%*30 : GET(10,YP)-(500,YP
-28),A%(0) : PUT(10,YP),A%(0),PRESET : W
INDOW %1 : DX=0 : DY=0
470 X=X+DX : Y=Y+DY : CURSOR POINT(X,Y)1
,3
480 IF MODO%=1 THEN PSET(X,Y),1 : GOTO 5
00
490 IF MODO%=3 THEN PSET(X,Y),0
500 GOSUB 1420 : IF K$="5" THEN MODO%=2
: GOTO 450
510 IF K$=CHR$(13) THEN 670
520 IF K$="8" AND Y<244-R THEN DX=0 : DY
=R : GOTO 470
530 IF K$="9" AND Y<244-R AND X<439-R TH
EN DX=R : DY=R : GOTO 470
```

OLIVETTI M20

Seguito listato M20 Paint.

```
540 IF K$="6" AND X<439-R THEN DX=R : DY
=0 : GOTO 470
550 IF K$="3" AND X<439-R AND Y>R THEN D
X=R : DY=-R : GOTO 470
560 IF K$="2" AND Y>R THEN DX=0 : DY=-R
: GOTO 470
570 IF K$="1" AND X>R AND Y>R THEN DX=-R
 : DY=-R : GOTO 470
580 IF K$="4" AND X>R THEN DX=-R : DY=0
: GOTO 470
590 IF K$="7" AND X>R AND Y<244-R THEN D
X=-R : DY=R : GOTO 470
600 IF K$="+" THEN R=10 : GOTO 470
610 IF K$="-" THEN R=1 : GOTO 470
620 IF K$=CHR$(46) THEN XMARK=X : YMARK=
Y : GOTO 660
630 IF K$=CHR$(158) AND MODO%>1 THEN MOD
O%=MODO%-1 : GOTO 450
640 IF K$=CHR$(156) THEN X=XMARK : Y=YMA
RK : GOTO 660
650 IF K$=CHR$(154) AND MODO%<8 THEN MOD
O%=MODO%+1 : GOTO 450
660 DX=0 : DY=0 : GOTO 470
670 IF MODO%<4 THEN 470
680 IF MODO%=4 THEN PAINT(X,Y),1 : GOTO
660
690 IF MODO%=5 THEN 730
700 IF MODO%=6 THEN 880
710 IF MODO%=7 THEN 1000
720 IF MODO%=8 THEN WINDOW %2 : PUT(10,Y
P),A%(0) : GOTO 180 ELSE 470
730 M$="Box? (y/n)" : GOSUB 1440 : L$="n
" : XK=13 : GOSUB 1450 : GOSUB 1710 : GO
SUB 1580 : IF L$="y" OR L$="Y" THEN 810
740 ON ACTV GOTO 750,760,770,780,790,800
750 LINE(X1,Y1)-(X2,Y2),,,AND : GOTO 110
0
760 LINE(X1,Y1)-(X2,Y2),,,XOR : GOTO 110
0
770 LINE(X1,Y1)-(X2,Y2),,,OR : GOTO 1100
780 LINE(X1,Y1)-(X2,Y2),,,NOT : GOTO 110
0
790 LINE(X1,Y1)-(X2,Y2),,,PSET : GOTO 11
00
800 LINE(X1,Y1)-(X2,Y2),,,PRESET : GOTO
1100
810 ON ACTV GOTO 820,830,840,850,860,870
820 LINE(X1,Y1)-(X2,Y2),,B,AND : GOTO 11
00
830 LINE(X1,Y1)-(X2,Y2),,B,XOR : GOTO 11
00
840 LINE(X1,Y1)-(X2,Y2),,B,OR : GOTO 110
0
850 LINE(X1,Y1)-(X2,Y2),,B,NOT : GOTO 11
00
860 LINE(X1,Y1)-(X2,Y2),,B,PSET : GOTO 1
100
870 LINE(X1,Y1)-(X2,Y2),,B,PRESET : GOTO
 1100
880 M$="Aspect ratio? " : GOSUB 1440 : X
K=16 : L$=" 0.807" : GOSUB 1450
890 IF VAL(L$)=0 THEN 9999 ELSE GOSUB 17
10 : GOSUB 1580
900 E=VAL(L$) : RD=SQR((X1-X2)^2+((Y1-Y2
)/E)^2) : IF E>.807 AND X1=X2 THEN RD=RD
*E
910 IF E>1 AND Y1=Y2 THEN RD=RD*E
920 ON ACTV GOTO 930,940,950,960,970.980
930 CIRCLE(X1,Y1),RD,,E,AND : GOTO 990
940 CIRCLE(X1,Y1),RD,,E,XOR : GOTO 990
950 CIRCLE(X1,Y1),RD,,E,OR : GOTO 990
960 CIRCLE(X1,Y1),RD,,E,NOT : GOTO 990
970 CIRCLE(X1,Y1),RD,,E,PSET : GOTO 990
980 CIRCLE(X1,Y1),RD,,E,PRESET
990 GOTO 1100
1000 GOSUB 1790 : GOSUB 1580 : GET(X1,Y1
)-(X2,Y2),S%(0)
1010 IF SC=1 THEN LINE(X1,Y1)-(X2,Y2),0,
BF
1020 GOSUB 1710 : M$="Indica la nuova po
sizione con il cursor point. " : GOSUB 1
440 : FLAG=0 : WINDOW %1 : GOSUB 1590
1030 ON ACTV GOTO 1040,1050,1060,1070,10
80,1090
1040 PUT(X2,Y2),S%(0),AND : GOTO 1100
1050 PUT(X2,Y2),S%(0),XOR : GOTO 1100
1060 PUT(X2,Y2),S%(0),OR : GOTO 1100
1070 PUT(X2,Y2),S%(0),NOT : GOTO 1100
1080 PUT(X2,Y2),S%(0),PSET : GOTO 1100
1090 PUT(X2,Y2),S%(0),PRESET
1100 WINDOW %2 : MODO%=2 : GOTO 450
1110 M$="Full screen£? (y/n) " : GOSUB 1
440 : XK=21 : L$="y" : GOSUB 1450
1120 IF L$="Y" OR L$="y" THEN ELEMAT=846
0 : WINDOW %1 : GET(1,243)-(438,1),S%(0)
 : GOTO 1150
1130 M$="Delimita la superficie da memor
izzare. " : GOSUB 1440
1140 FOR TIME=1 TO 1000 : NEXT TIME : GO
SUB 1580 : GET(X1,Y1)-(X2,Y2),S%(0) : EL
EMAT=(FIX(ABS(X2-X1)/16-.01)+1)*ABS(Y2-Y
1)+3
1150 M$="File name?" : GOSUB 1440 : XK=1
4 : L$=SPACE$(14) : GOSUB 1450
1160 OPEN "O",#1,L$
1170 FOR J=0 TO ELEMAT-1 STEP 10 : FOR K
=J TO J+8 : PRINT #1,S%(K); : NEXT K : P
RINT #1,S%(J+9) : NEXT J
1180 CLOSE #1 : GOSUB 1430 : GOTO 180
1190 M$="Stampa del bordo? (y/n) " : GOS
UB 1440
1200 XK=26 : L$="y" : GOSUB 1450
1210 IF L$="n" OR L$="N" THEN WINDOW %1
: LINE(0,0)-(439,244),0,B
1220 M$="E' collegata la stampante? (y/n
)" : GOSUB 1440
1230 XK=35 : L$="y" : GOSUB 1450
1240 IF L$="y" OR L$="Y" THEN CLOSE WINDOW %2 : CL
```

Seguito listato M20 Paint.

```
OSE WINDOW %3 : CURSOR(1,1)0 :CALL "sp" ELSE WINDO
W %1 : LINE(0,0)-(439,244),,B : GOTO 190
1250 RESTORE 1850 : W2=WINDOW(2,7) : W3=
WINDOW(1,245) : GOTO 130
1260 M$="Tutto lo schermo? (y/n)" : GOSU
B 1440 : XK=26 : L$="y" : GOSUB 1450
1270 IF L$="y" OR L$="Y" THEN X1=1 : Y1=
1 : X2=438 : Y2=243 : WINDOW %1 : GOTO 1
300
1280 M$="Delimita la superficie da cance
llare. " : GOSUB 1440
1290 FOR TIME=1 TO 1000 : NEXT TIME : GO
SUB 1580
1300 LINE(X1,Y1)-(X2,Y2),0,BF : GOTO 180
1310 XT=2 : YT=2 : M$="Modo testo." : GO
SUB 1440 : WINDOW %1
1320 CURSOR(XT,YT)1,1 : GOSUB 1420 : IF
ASC(K$)<32 OR ASC(K$)>127 THEN 1350
1330 CURSOR(XT,YT)1,1 : PRINT K$; : IF X
T<72 THEN XT=XT+1 : GOTO 1320
1340 IF YT<23 THEN YT=YT+1 : XT=2 :GOTO
1320 ELSE GOTO 1320
1350 IF K$=CHR$(13) THEN CURSOR(1,1)0 :
GOTO 180
1360 IF K$=CHR$(154) AND YT<23 THEN YT=Y
T+1 : GOTO 1320
1370 IF K$=CHR$(155) AND XT>2 THEN XT=XT
-1 : GOTO 1320
1380 IF K$=CHR$(157) AND XT<72 THEN XT=X
T+1 : GOTO 1320
1390 IF K$=CHR$(158) AND YT>2 THEN YT=YT
-1 : GOTO 1320 ELSE 1320
1400 M$=" * ERRORE * PREMI UN TASTO PER
CONTINUARE " : GOSUB 1440 : GOSUB 1420
1410 WINDOW %2 : PUT(10,YP),A%(0) : RESU
ME 180
1420 K$=INKEY$ : IF K$="" THEN 1420 ELSE
 RETURN
1430 WINDOW %3 : LINE(0,0)-(511,255),,BF
 : RETURN
1440 GOSUB 1430 : COLOR 0,1 : CURSOR(2,1
) : PRINT M$; : COLOR 1,0 : RETURN
1450 XX=XK : LS=LEN(L$) : WINDOW %3 : CO
LOR 0,1 : CURSOR(XK,1) : PRINT L$;
1460 CURSOR(XX,1)1,1 : GOSUB 1420 : IF A
SC(K$)<32 OR ASC(K$)>127 THEN 1510
1470 IF XX=XK THEN L$=K$+RIGHT$(L$,LS-1)
 : GOTO 1500
1480 IF XX=XK+LS-1 THEN L$=LEFT$(L$,LS-1
)+K$ : GOTO 1500
1490 L$=LEFT$(L$,XX-XK)+K$+RIGHT$(L$,LS-
XX+XK-1)
1500 PRINT K$; : IF XX<XK+LS-1 THEN XX=X
X+1 : GOTO 1460 ELSE GOTO 1460
1510 IF K$=CHR$(8) AND XX>XK THEN XX=XX-
1 : CURSOR(XX,1) : PRINT " "; : GOTO 146
0
1520 IF K$=CHR$(155) AND XX>XK THEN XX=X
X-1 : GOTO 1460
1530 IF K$=CHR$(157) AND XX<LS+XK-1 THEN
 XX=XX+1 : GOTO 1460
1540 IF K$<>CHR$(13) THEN 1460
1550 FOR J=LS TO 1 STEP -1 : IF RIGHT$(L
$,1)<>CHR$(32) THEN 1570
1560 L$=LEFT$(L$,LEN(L$)-1) : NEXT J
1570 COLOR 1,0 : RETURN
1580 FLAG=1 : X1=X : X2=X : Y1=Y : Y2=Y
: M$="Indica il primo punto con il curso
r point. " : GOSUB 1440 : WINDOW %1
1590 CURSOR POINT(X,Y)1,3 : GOSUB 1420 :
 IF K$="1" AND X>R AND Y>R THEN X=X-R :
Y=Y-R : GOTO 1590
1600 IF K$="2" AND Y>R THEN Y=Y-R : GOTO
 1590
1610 IF K$="3" AND X<439-R AND Y>R THEN
X=X+R : Y=Y-R : GOTO 1590
1620 IF K$="4" AND X>R THEN X=X-R : GOTO
 1590
1630 IF K$="6" AND X<439-R THEN X=X+R :
GOTO 1590
1640 IF K$="7" AND X>R AND Y<244-R THEN
X=X-R : Y=Y+R : GOTO 1590
1650 IF K$="8" AND Y<244-R THEN Y=Y+R :
GOTO 1590
1660 IF K$="9" AND X<439-R AND Y<244-R T
HEN X=X+R : Y=Y+R : GOTO 1590
1670 IF K$="+" THEN R=10 : GOTO 1590
1680 IF K$="-" THEN R=1 : GOTO 1590
1690 IF K$=CHR$(13) AND FLAG=1 THEN X1=X
 : Y1=Y : FLAG=2 : M$="Indica il secondo
punto con il cursor point. " : GOSUB 14
40 : WINDOW %1 : GOTO 1590
1700 IF K$=CHR$(13) THEN X2=X : Y2=Y : G
OSUB 1430 : WINDOW %1 : RETURN ELSE 1590
1710 M$="Action verb:  AND   XOR   OR
NOT   PSET  PRESET " : GOSUB 1440
1720 ACTV=5 : W=0
1730 COLOR 0,1 : XP=10+ACTV*6 : CURSOR(X
P,1) : PRINT BO$(ACTV);
1740 XP=XP+W*6 : COLOR 1,0 : CURSOR(XP,1
)0 : ACTV=ACTV+W : PRINT BO$(ACTV);
1750 GOSUB 1420 : IF K$="4" AND ACTV>1 T
HEN W=-1 : GOTO 1730
1760 IF K$="6" AND ACTV<6 THEN W=1 : GOT
O 1730
1770 IF K$=CHR$(13) THEN GOSUB 1430 : RE
TURN
1780 GOTO 1750
1790 GOSUB 1430 : M$="Sposta  Copia" : G
OSUB 1440
1800 CURSOR(2,1)0 : PRINT "Sposta "; : S
C=1
1810 GOSUB 1420 : IF K$="5" OR K$=CHR$(1
3) THEN GOSUB 1430 : RETURN
1820 IF K$="4" AND SC=2 THEN COLOR 0,1 :
 CURSOR(10,1) : PRINT "Copia "; : COLOR
1,0 : GOTO 1800
1830 IF K$="6" AND SC=1 THEN SC=2 : COLO
```

Seguito listato M20 Paint.

```
R 0,1 : CURSOR(2,1) : PRINT "Sposta "; :
 COLOR 1,0 : CURSOR(10,1) : PRINT "Copia
"; : GOTO 1820
1840 GOTO 1810
1850 DATA 38,29,256,0,0,0,32767,-1,-2048
,16384,0,2048,16385,-256,2048,16386
1860 DATA 128,2048,16388,6208,2048,16392
,15392,2048,16396,6240,2048,16394,160
1870 DATA 2048,16393,-224,2048,16393,288
,2048,16393,288,2048,16393,288,2048
1880 DATA 16393,288,2048,16393,288,2048,
16393,288,2048,16393,288,2048,16393
1890 DATA 288,2048,16393,288,2048,16389,
320,2048,16387,-128,2048,16385,256,2048
1900 DATA 16384,-32256,2048,16384,31744,
2048,16384,14336,2048,16384,4096,2048
1910 DATA 16384,0,2048,32767,-1,-2048,0,
0,0
1920 DATA 38,29,256,0,0,0,32767,-1,-2048
,16384,0,2048,16384,0,2048,16384,0
1930 DATA 2048,16384,0,2048,16384,0,2048
,16384,0,2048,16384,0,2048,16415,-8
1940 DATA 2048,16424,4,2048,16452,2,2048
,16515,-1,2048,16834,1,2048,17346,1
1950 DATA 2048,16834,1,2048,16515,-1,204
8,16452,2,2048,16424,4,2048,16415,-8
1960 DATA 2048,16384,0,2048,16384,0,2048
,16384,0,2048,16384,0,2048,16384,0
1970 DATA 2048,16384,0,2048,16384,0,2048
,32767,-1,-2048,0,0,0
1980 DATA 38,29,256,0,0,0,32767,-1,-2048
,16384,0,2048,16384,0,2048,16384,0
1990 DATA 2048,16384,32767,2048,16385,-2
,-30720,16387,-3,-30720,16391,-5
2000 DATA -30720,16399,-9,-30720,16400,3
1,-30720,16416,47,-30720,16448,79
2010 DATA -30720,16512,143,2048,16640,27
1,2048,16895,-498,2048,16896,524,2048
2020 DATA 16896,520,2048,16896,528,2048,
16896,544,2048,16896,576,2048,16896,640
2030 DATA 2048,16896,768,2048,16895,-102
4,2048,16384,0,2048,16384,0,2048,16384,0
2040 DATA 2048,32767,-1,-2048,0,0,0
2050 DATA 38,29,256,0,0,0,32767,-1,-2048
,16384,0,2048,16384,6144,2048,16384
2060 DATA 9216,2048,16384,16896,2048,163
84,16896,2048,16384,16896,2048,16384
2070 DATA 16896,2048,16384,16896,2048,16
384,9216,2048,16384,9216,2048,16384
2080 DATA 9216,2048,16384,15360,2048,163
84,15360,2048,16447,-8,2048,16416,8
2090 DATA 2048,16416,8,2048,16416,8,2048
,16447,-8,2048,16415,-16,2048,16415
2100 DATA -16,2048,16405,21840,2048,1640
5,21840,2048,16405,21840,2048,16384
2110 DATA 0,2048,16384,0,2048,32767,-1,-
2048,0,0,0
2120 DATA 38,29,256,0,0,0,32767,-1,-2048
,16384,0,2048,16384,0,2048,16384,0
2130 DATA 2048,16384,0,2048,16384,0,2048
,16415,-64,2048,16400,192,2048,16400
2140 DATA 320,2048,16400,576,2048,16400,
1088,2048,16400,2112,2048,16400,4160
2150 DATA 2048,16400,8256,2048,16400,164
48,2048,16400,-32704,2048,16401,64
2160 DATA 2048,16402,64,2048,16404,64,20
48,16408,64,2048,16415,-64,2048,16384
2170 DATA 0,2048,16384,0,2048,16384,0,20
48,16384,0,2048,16384,0,2048,32767
2180 DATA -1,-2048,0,0,0
2190 DATA 38,29,256,0,0,0,32767,-1,-2048
,16384,0,2048,16384,0,2048,16384,0
2200 DATA 2048,16384,-512,2048,16387,384
,2048,16396,96,2048,16432,24,2048,16416
2210 DATA 8,2048,16448,4,2048,16448,4,20
48,16512,2,2048,16512,2,2048,16512,2
2220 DATA 2048,16512,2,2048,16512,2,2048
,16448,4,2048,16448,4,2048,16416,8,2048
2230 DATA 16432,24,2048,16396,96,2048,16
387,384,2048,16384,-512,2048,16384,0
2240 DATA 2048,16384,0,2048,16384,0,2048
,32767,-1,-2048,0,0,0
2250 DATA 38,29,256,0,0,0,32767,-1,-2048
,16384,0,2048,16384,0,2048,16384,0
2260 DATA 2048,16384,0,2048,16384,2048,2
048,16384,3072,2048,16384,3584,2048
2270 DATA 16384,3840,2048,16384,3968,204
8,16415,-64,2048,16415,-32,2048,16415
2280 DATA -16,2048,16415,-8,2048,16415,-
16,2048,16415,-32,2048,16415,-64,2048
2290 DATA 16384,3968,2048,16384,3840,204
8,16384,3584,2048,16384,3072,2048
2300 DATA 16384,2048,2048,16384,0,2048,1
6384,0,2048,16384,0,2048,16384,0,2048
2310 DATA 32767,-1,-2048,0,0,0
2320 DATA 38,29,256,0,0,0,32767,-1,-2048
,16384,0,2048,16384,0,2048,16384,0
2330 DATA 2048,16384,0,2048,16384,0,2048
,16384,0,2048,16384,0,2048,16384,0
2340 DATA 2048,16384,0,2048,16508,14392,
2048,16448,16452,2048,16448,16448,2048
2350 DATA 16504,14400,2048,16448,1088,20
48,16448,1092,2048,16508,30776,2048
2360 DATA 16384,0,2048,16384,0,2048,1638
4,0,2048,16384,0,2048,16384,0,2048
2370 DATA 16384,0,2048,16384,0,2048,1638
4,0,2048,16384,0,2048,32767,-1,-2048
2380 DATA 0,0,0
```

Prospettiva e grafica tridimensionale

Nell'articolo apparso sul n. 32 - Ottobre '85 - nelle righe 2320 e 2420 manca il simbolo π dopo l'asterisco.

Life

Anche per il QL, uno dei più famosi giochi di simulazione

di Francesco Balena

Un breve cenno su Life per quei lettori che non ne hanno mai sentito parlare: si tratta di un gioco di simulazione inventato quasi trent'anni fa dal matematico americano Conway in cui si assiste all'evolversi nel tempo di una alquanto improbabile colonia di cellule. Questa colonia è improbabile perché si estende esclusivamente nel piano (non conosce la terza dimensione) e soprattutto perché le sue cellule nascono e muoiono seguendo delle rigide regole matematiche; nonostante ciò, Life mantiene un fascino tutto particolare, soprattutto per l'imprevedibilità delle configurazioni che possono venirsi a formare, anche partendo da una colonia di poche cellule.

Le regole che determinano la vita della colonia sono davvero molto semplici: ogni locazione nel piano contiene una cellula, che può essere viva oppure morta (oppure potete dire che contiene una cellula o non contiene niente, se non volete assolutamente ammettere che le cellule possano risorgere); una cellula morta torna a essere viva se confina con esattamente tre cellule vive (si considerano adiacenti le otto locazioni che circondano quella posizione), mentre una cellula viva può continuare a vivere se ha due oppure tre cellule vicine vive; se invece ha meno di due oppure più di tre vicini vivi essa muore (per isolamento nel primo caso e per soffocamento nel secondo).

Per non porre dei limiti alla crescita della colonia, Life dovrebbe svolgersi su un piano infinito, cosa chiaramente impossibile adoperando il computer; spesso in casi come questo si ricorre all'artificio di considerare adiacenti il bordo inferiore e quello superiore, come pure i due bordi laterali (il risultato è quello che si avrebbe giocando su una superficie toroidale): il piano di gioco rimane finito, ma in questo modo ogni cellula continua ad avere sempre otto vicini e si evitano fastidiose distorsioni ai limiti del campo.

Anche questo programma di Life per il QL adotta questo artificio per espandere logicamente il campo, che misura 24 righe per 30 colonne; è scritto esclusivamente in SuperBasic, ma è lo stesso molto veloce (una decina di secondi per colonie di una trentina di elementi), grazie a un algoritmo ottimizzato e a qualche trucchetto che spiegheremo più avanti.

Dando il Run al programma ci si porta in fase di creazione della colonia: possiamo spostarci sul piano con le frecce e lasciar cadere delle cellule vive con la barra spaziatrice, che può anche essere usata per correggere gli errori. Come spiega anche il messaggio nella finestra in basso a destra, in questa fase sono attivi altri due tasti: F1 che pulisce l'intero schermo (in pratica effettua un Run) e F2 che termina la fase di creazione e che potremo usare solo quando saremo soddisfatti della configurazione iniziale per passare alla fase di elaborazione vera e propria.

A questo punto, dopo che è stata calcolata e visualizzata la prima generazione di cellule, abbiamo la possibilità di visualizzare le generazioni successive premendo un tasto qualsiasi, a eccezione di F1 che ci rimanda alla fase di creazione (o meglio di editing, in questo caso) e di F2 che attiva l'auto-mode, in cui le generazioni vengono create senza attendere l'intervento dell'utente; per uscire dall'auto-mode è sufficente premere un qualsiasi tasto.

Il programma

Scrivere un programma come Life sembra a prima vista molto facile (e in parte lo è): basta considerare una matrice bidimensionale che rappresenta le cellule della colonia e per ogni generazione calcolare il numero di vicini vivi per ogni posizione e decidere così per la vita o per la morte di ogni cellula.

Eppure questo metodo non funziona correttamente. Infatti poiché si suppone che le nascite e morti avvengono nello stesso istante per tutte le cellule, avremo bisogno di *due* matrici, una per tener conto della situazione attuale (su cui contare il numero dei vicini) e l'altra per memorizzare la situazione che si verrà a creare all'inizio della generazione successiva. Tra una generazione e l'altra dovremo ricordarci di caricare nella prima matrice gli elementi della seconda, con una perdita di tempo non indifferente.

Quest'ultimo sistema funziona, ma risulta un po' troppo lento, perché la matrice deve essere esaminata tutta, anche se la colonia conta una minima parte di cellule vive.

Il presente programma utilizza invece un approccio alquanto diverso. Le matrici in gioco sono ancora due: la prima memorizza la situazione della colonia (uno per le cellule vive e zero per quelle morte) mentre l'altra conta per ciascuna posizione il numero dei suoi vicini vivi e viene aggiornata a ogni nascita o morte, esattamente come la prima. A esempio, se una cellula risorge, il programma memorizzerà uno nella posizione corrispondente nella prima matrice e incrementerà di uno tutte le caselle della seconda matrice adiacenti a quella della cellula appena nata. Il vantaggio di questo sistema è evidente: nel primo metodo il calcolo dei vicini veniva ripetuto per tutte le 24 x 30 = 720 caselle, mentre così viene effettuato solo per quelle (poche) cellule che cambiano stato da una generazione all'altra.

In realtà il programma usa *due* matrici per memorizzare il numero dei vicini, esattamente come nel metodo prima descritto. In questo caso occorre usare due matrici per memorizzare la situazione della colonia: una per la situazione attuale e l'altra per la situazione che si

verrà a creare all'inizio della generazione successiva.

L'altro trucco usato dal programma è quello di impiegare non delle matrici bidimensionali, bensì delle semplici stringhe, e cioè C$ per la situazione della colonia (conterrà dei caratteri zero oppure uno a seconda che la cellula corrispondente sia morta o viva), p$ o n$ per contare il numero dei vicini vivi per ciascuna posizione (che conterranno dei numeri da zero a otto), ovviamente tutte e tre della stessa dimensione. Perché usare delle stringhe e non delle matrici bidimensionali numeriche (che tra l'altro renderebbero il programma più leggibile)? Perché di tutte le cellule del campo a noi interessano solo quelle vive, che possono morire se non hanno due o tre vicini vivi, e quelle con esattamente tre vicini vivi, che se sono morte possono tornare a vivere. Poiché queste cellule sono in genere una piccola parte di tutte le cellule del gioco, è molto conveniente fermare la nostra attenzione su di esse soltanto, rintracciandole nelle stringhe sopra citate per mezzo della velocissima funzione di Instring (vedere riga 1990) ed evitando di prendere in considerazione tutte le cellule morte che sicuramente rimarranno tali nella prossima generazione.

Un altro vantaggio niente affatto trascurabile (in termini di tempi di elaborazione) viene dal fatto che per trasferire il contenuto di una matrice in un'altra, occorre trasferire ogni singolo elemento, mentre nel caso di due stringhe, il tutto si risolve con una semplice e rapidissima assegnazione (riga 1840).

Principali variabili globali

nri - Numero delle righe del campo di gioco (24).
nco - Numero delle colonne del campo di gioco (30).
cell - Numero delle cellule vive.
gener - Numero di generazioni elaborate.
c$ - Stringa di caratteri che rappresenta la situazione della colonia (uno per le cellule vive e zero per quelle morte).
p$ - Stringa che conta il numero dei vicini per ogni posizione del campo; essa viene testata nella procedura "genera" insieme alla precedente per decidere il destino di ogni cellula.
**n$** - Anche questa stringa conta il numero dei vicini per ogni cellula, ma a differenza della p$, viene aggiornata man mano che le cellule cambiano stato, per tener conto della situazione che si verrà a creare all'inizio della generazione seguente.
Nota: Tutte queste tre stringhe non contano 720 elementi, come ci si potrebbe aspettare, bensì 26 x 32 = 832 caratteri; derivano cioè da una matrice con due righe e due colonne in più, per tener conto del fatto che i bordi opposti sono considerati adiacenti e per evitare fastidiosi controlli sugli indici.
co1, co2, co3 - coefficienti (calcolati una volta per tutte nelle righe 1370-1380) che stabiliscono la corrispondenza tra il campo di gioco (quello che vediamo sullo schermo) e le stringhe che lo rappresentano internamente. A esempio, l'elemento (r,c) della matrice corrisponde all'elemento:

$$r \star co1 + c + 1$$

di una delle tre stringhe c$, p$ o n$.

Listato 1 - Il programma Life.

```
1000 REMark -------------------------
        L I F E
                                      Fra
ncesco  Balena - 1985
                                ------------
-------------------
1010 init
1020 crea
1030 REPeat loop
1040  CLS #7: INK #7,2: FLASH #7,1
1050  PRINT #7,\\\\\' ELABO RAZIONE':
  FLASH #7,0
1060  genera
1070  BEEP 800,6
1080  CLS #7: INK #7,4: PRINT #7\\
   ' PREMI   UN    TASTO'
1090  INK #7,7: PRINT #7\\\'F1 PER RESTA
RT'\\'F2 PER AUTOGEN'
1100  RANDOMISE KEYROW(0)
1110  tasto=CODE(INKEY$(-1))
1120  IF tasto=232: crea: NEXT loop
1130  IF tasto=236 THEN
1140   CLS #7: INK #7,2: FLASH #7,1
1150   PRINT #7\\\' AUTO   MODE'
1160   FLASH #7,0: INK #7,7: PRINT
   #7\\\' PREMI   UN    TASTO'
1170   REPeat cont: genera:BEEP 800,6
   : IF INKEY$<>'' THEN EXIT cont
1180  END IF
1190 END REPeat loop
1200 REMark -------------------------
           INIZIALIZZAZIONE
------------------------------
1210 DEFine PROCedure init
1220 LOCal j,k
1230 MODE 8
1240 OPEN #4,scr_512x256a0x0
1250 PAPER #4,0,1,0: CLS #4
1260 OPEN #4,scr_364x242a25x10
1270 PAPER #4,0:INK #4,7:OVER #4,-1:
  CLS #4: BORDER #4,1,2
1280 OPEN #5,scr_64x24a410x34
1290 PAPER #5,0: CLS #5:BORDER #5,1,7
1300 INK #5,7:STRIP #5,2:PRINT #5,
  'GENER':STRIP #5,0
1310 OPEN #6,scr_64x24a410x75
1320 PAPER #6,0: CLS #6:BORDER #6,1,7
1330 INK #6,7:STRIP #6,2: PRINT #6,
  'CELLS':STRIP #6,0
1340 OPEN #7,scr_88x126a398x120
```

Seguito listato Life.

```
1350 PAPER #7,0:CLS #7:BORDER #7,1,7
1360 nri=24: nco=30
1370 co1=nco+2: co2=nri*co1
1380 co3=(nri+1)*co1
1390 cell=0
1400 c$= FILL$('0',(nri+2)*co1)
1410 n$=c$: p$=c$
1420 END DEFine init
1430 REMark -----------------------
     CREAZIONE DELLA 'COLONIA'      -
------------------------------
1440 DEFine PROCedure crea
1450 LOCal r,c,cur$
1460 gener=0
1470 CLS #7: OVER #4,0
1480 INK #7,4:PRINT #7,' USA I  ┘┘┐┘'
1490 INK #7,7:PRINT #7\\'<SPACE>  PER  O
N/OFF'\\'F1 PER  CLEAR'\\'F2 PER FINIRE'
1500 r=nri DIV 2: c=nco DIV 2
1510 REPeat sposta
1520  cur$='+': IF c$(r*co1+c+1)=1
  THEN cur$='O'
1530  AT #4,r-1,c-1: PRINT #4,cur$
1540  tasto=CODE(INKEY$(5))
1550  AT #4,r-1,c-1:PRINT #4,' '
1560  IF cur$='O' THEN AT #4,r-1,c-1
  : PRINT #4,cur$
1570  SELect ON tasto
1580   ON tasto=192
1590    c=c-1:IF c=0 THEN c=nco
1600   ON tasto=200
1610    c=c+1:IF c>nco THEN c=1
1620   ON tasto=208
1630    r=r-1:IF r=0 THEN r=nri
1640   ON tasto=216
1650    r=r+1:IF r>nri THEN r=1
1660   ON tasto=32
1670    BEEP 200,3
1680    st=1-c$(r*co1+c+1)*2
1690    aggiorna r*co1+c+1,st
1700   ON tasto=232
1710    RUN
1720   ON tasto=236
1730    EXIT sposta
1740  END SELect
1750 END REPeat sposta
1760 OVER #4,-1
1770 END DEFine modifica
1780 REMark -----------------------
          NUOVA  GENERAZIONE
------------------------------
1790 DEFine PROCedure genera
1800 LOCal r,c,st,el,succ
1810 gener=gener+1
1820 AT #5,1,0: PRINT #5,FILL$(' ',4-
  LEN(gener));gener;
1830 AT #6,1,0: PRINT #6,FILL$(' ',4-
  LEN(cell));cell;
1840 p$=n$
1850 FOR r=nco+3 TO LEN(p$) STEP co1
1860  p$(r+1)=n$(r+1)+p$(r+nco+1)
1870  p$(r+nco+1)=0
1880  p$(r+nco)=p$(r+nco)+p$(r)
1890  p$(r)=0
1900 END FOR r
1910 FOR c=2 TO nco+1
1920  p$(co1+c)=p$(co1+c)+p$(co3+c)
1930  p$(co3+c)=0
1940  p$(co2+c)=p$(co2+c)+p$(c)
1950  p$(c)=0
1960 END FOR c
1970 el=nco+3
1980 REPeat cerca
1990  succ=primo('1' INSTR c$(el+1 TO
   ),'3' INSTR p$(el+1 TO ))
2000   IF succ=0 THEN EXIT cerca
2010  el=el+succ
2020   IF c$(el)=1 THEN
2030    IF p$(el)<>2 AND p$(el)<>3
   THEN aggiorna el,-1
2040   ELSE
2050    IF p$(el)=3 : aggiorna el,1
2060   END IF
2070 END REPeat cerca
2080 END DEFine genera
2090 REMark -----------------------
          AGGIORNA I VETTORI PER
       UNA 'NASCITA' O 'MORTE'      ----
--------------------------
2100 DEFine PROCedure aggiorna (el,st)
2110 c$(el)=c$(el)+st
2120 cell=cell+st
2130 n$(el-33)=n$(el-33)+st
2140 n$(el-32)=n$(el-32)+st
2150 n$(el-31)=n$(el-31)+st
2160 n$(el-1)=n$(el-1)+st
2170 n$(el+1)=n$(el+1)+st
2180 n$(el+31)=n$(el+31)+st
2190 n$(el+32)=n$(el+32)+st
2200 n$(el+33)=n$(el+33)+st
2210 AT #4,el DIV co1-1,el MOD co1-2
2220 PRINT #4,'O'
2230 END DEFine aggiorna
2240 REMark -----------------------
          FUNZIONE 'PRIMO'
--------------------------
2250 DEFine FuNction primo(a,b)
2260 IF b=0 OR (a<>0 AND a<b)THEN
  RETurn a
2270 RETurn b
2280 END DEFine primo
```

Commento al listato

1000-1020 - Si inizializzano variabili e finestre e si permette di creare la configurazione iniziale di cellule.
1030-1110 - Ciclo principale del programma; viene visualizzata la generazione successiva e si attende la pressione di un tasto. Notare l'istruzione in riga 1100 che ha l'effetto di svuotare il buffer di tastiera.
1120-1190 - A seconda di quale tasto è stato premuto si torna alla fase di edit (F1), si attiva l'auto-mode (F2) o semplicemente si elabora la generazione seguente. Notate che non c'è modo di uscire dal ciclo principale, e il programma si può arrestare solo dando il Break o il Reset.
1200-1420 - Inizializzazione delle finestre e delle variabili usate dal programma.
1430-1500 - Procedure di creazione della colonia; la stessa procedura è richiamata a gioco iniziato per effettuare delle piccole variazioni. La variabile cur$ conterrà il simbolo con cui verrà rappresentato il cursore lampeggiante (cioè " + " oppure 0 a seconda che si trovi su una casella libera o meno).
1510-1770 - Loop di attesa; notate la Select in riga 1570 che distingue le varie azioni da compiere in risposta alla pressione di un determinato tasto. Le variabili r e c memorizzano la posizione del cursore.
1780-1830 - È la Procedura che elabora una nuova generazione di cellule, ed è quindi un po' il cuore di Life. Notate come vengono stampate giustificate a destra le informazioni nelle finestre #5 e #6.
1840-1960 - La stringa p$ contiene il numero dei vicini vivi per ciascuna posizione. I due cicli For/End For aggiornano gli elementi di p$ che si riferiscono alle cellule del bordo (i cui vicini si trovano quindi sul lato opposto).
1970-2080 - Ciclo in cui vengono ricercate le cellule vive che devono morire e viceversa; "el" punta all'elemento in esame. Notate come in riga 1990 vengono analizzate soltanto le cellule vive (il cui elemento in c$ è uno) e le cellule con esattamente tre vicini, facendo uso della funzione Instring e Primo.
2090-2230 - Procedura di aggiornamento, che viene richiamata in caso di nascita o morte di una cellula; "el" punta all'elemento da aggiornare e st vale +1 o –1 (rispettivamente per una nascita e una morte). Alle righe 2130-2200 viene incrementato (o decrementato) il contatore dei vicini vivi per tutte le cellule che circondano la cellula nata (o morta).
L'istruzione in riga 2220 funziona correttamente perché nella finestra #4 è attiva l'opzione di Over –1.
2240-2280 - La funzione Primo restituisce il minore tra due positivi, oppure il maggiore se uno dei due è nullo. ■

Premessa

Immaginate di avere un amico che fa il campanaro di professione. Ora provate a pensare di doverlo sostituire: anche i campanari vanno in ferie! Bene, se ve la sentite, copiate i listati e cominciate a giocare!

Il gioco

Le campane sono otto, ognuna contrassegnata da un numero posto sopra di esse.
Scopo del gioco è di riuscire a ripetere esattamente le sequenze di note che compaiono (e vengono suonate) all'inizio di ogni turno. Per far questo dovrete muovere il vostro alter-ego lungo lo schermo tramite le frecce: una volta giunti sotto la campana desiderata dovrete premere il numero corrispondente e l'omino, spiccando un balzo, la farà suonare.

Il campanaro

Tra bit, campane e farfalle, probabilmente "impazzirete" nel cimentarvi in questo gioco

di Ugo Rossini

Purtroppo non sarete soli: infatti nella torre campanaria vive un topo fastidioso che ha la cattiva abitudine di venirvi incontro per farvi cadere. Per cacciarlo dovrete andare a tirare la corda del fischietto contrassegnato da zero: gli ultrasuoni lo spaventeranno facendolo fuggire.

Inoltre dovete sapere che la torre campanaria è situata in mezzo alle colline toscane, e che tra i fiori svolazzano felicemente molte farfalle. Da una finestra aperta, ogni tanto ne entra una che si va a posare sulla corda di una campana facendola suonare (il meccanismo è molto sensibile!): se, fortunatamente, si

Listato 1 - Il programma Il campanaro.

```
10  PRINT  CHR$ (4)"BLOAD LETTORE
     .SUONI"
20  PRINT  CHR$ (4)"BLOAD CAM.SUO
     NI"
30  PRINT  CHR$ (4);"BLOAD CAM.SH
     APE"
40  POKE 232,0: POKE 233,64
50  HOME : PRINT ".......IL CAMPA
     NARO.......BY UG@......."
60  POKE  - 16368,0: VTAB 10: INPUT
     "IL TUO NOME ";NO$: IF NO$ =
     "" THEN 60
70  HOME : VTAB 23: PRINT "LIVELL
     O 1"; TAB( 20);"PUNTI 0"
80 NP = 2:LI = 1
90  HGR
100  HCOLOR= 3: SCALE= 1: ROT= 0
110  HPLOT 1,1 TO 1,143 TO 278,14
     3 TO 278,141 TO 3,141 TO 3,1

120  FOR J = 1 TO 22: READ A,B,C:
     DRAW A AT B,C: NEXT
130  FOR J = 1 TO 10: READ A: HPLOT
     A,60 TO A,111: NEXT
140 A = 0:B = 0:C = 0
150 X = 35:X1 = X
160  XDRAW  NOT B + BB + 17 AT X *
     5,130
170 NP = NP + 1: IF NP = 8 THEN N
     P = 3:LI = LI + 1: VTAB 23: HTAB
     9: PRINT LI: CALL 768,11,5,1
     ,10 - LJ: FOR K = 1 TO 500: NEXT

180 LJ = LI: IF LJ > 9 THEN LJ =
     9
200  REM  SCELTA NOTE
210  INVERSE
220  FOR M = 1 TO NP
230 N(M) =  INT (8 *  RND (1) + 1
     )
240  VTAB 21: HTAB M * 2 - 1: PRINT
     N(M)
250  CALL 768,N(M),3,1,11 + N(M) *
     2
260  FOR K = 1 TO 500: NEXT
270  NEXT
280  NORMAL
290  VTAB 21: SPEED= 1: PRINT  SPC(
     NP * 2): SPEED= 255
300  REM  MAIN
310 TA =  PEEK ( - 16384)
```

tratta della campana giusta nella sequenza richiesta, potete proseguire; ma se, come è molto più probabile, si tratta di una campana errata, verrete eliminati dal gioco. È consigliabile quindi spaventare la farfalla: tirate la corda del fischietto numero nove, ed essa scapperà! Seguite rigorosamente la sequenza stabilita, e non tirate mai le corde dei fischietti se non c'è presente l'animale indesiderato: il suono, infatti, non trovando il bersaglio cercato, si scaricherebbe su di voi, eliminandovi.

I livelli

La prima sequenza è di tre campane, alle quali, a ogni turno, se ne aggiunge una, fino ad arrivare a sette, dopodiché si ripartirà con tre campane al livello successivo.
A ogni livello superato aumenta la frequenza dei topi e delle farfalle, con conseguente maggior disagio (per non dir di peggio!).

I punti

Ogni campana suonata (al momento giusto) assegna un punteggio. Le campane più grosse (e più lontane dai fischietti) valgono di più. Inoltre i valori delle campane aumentano con l'incremento dei livelli. All'inizio del gioco viene richiesto il vostro nome per l'inserimento del punteggio (se ci riuscite) nella "Top ten".

Caricamento del programma

Scrivete la tavola delle figure (Cam.Shape) e il programma (Il campanaro). Scrivete anche il programma "Lettore di suoni" e la relativa tabella dei "Suoni": il programma, infatti, sfrutta l'ottimo generatore di suoni pubblicato su *Applicando* n. 15.
La prima volta che giocate, inoltre, date il Run anche al programma Creaz Cam.Rec che creerà un file (Cam.Rec) utilizzato dal programma principale per registrare i dieci migliori risultati di tutti i tempi.

Suggerimenti

1) Il vostro alter-ego è meno lento degli avversari, ma non sopravvalutate troppo questa caratteristica (soprattutto per quanto riguarda la farfalla: infatti questa si potrebbe posare anche sulla prima campana, rendendo vana la vostra "corsa" verso il fischietto).
2) Quando vi viene mostrata la nuova sequenza, il gioco è momentaneamente sospeso (e quindi restate fermi voi, il topo e la farfalla).
3) Ripetete continuamente tra di voi la sequenza dei numeri: è facilissimo dimenticarla mentre si sta sfuggendo a un nemico.
4) A volte (soprattutto ai livelli alti) è meglio non allontanarsi troppo dai fischietti: per le ultimissime campane (le più grosse) si può giocare sull'attesa, sperando in una farfalla...amica! ■

Seguito listato Il campanaro.

```
320  IF TA > 175 AND TA < 186 THEN
      GOSUB 600: GOTO 310
330  IF WZ THEN  GOSUB 1200: GOTO
     360
340  IF  RND (1) < LJ / 120 THEN
     WZ = 1
350  FOR J = 1 TO 40: NEXT
360  IF WW THEN  GOSUB 1000: GOTO
     390
370  IF  RND (1) < LJ / 100 THEN
     WW = 1
380  FOR J = 1 TO 40: NEXT
390  IF RI THEN RI = 0: GOTO 170
400  IF TA <  > 136 AND TA <  > 1
     49 THEN 310
410  IF X <  = 5 THEN X = 6:TA =
     149
420  IF X >  = 55 THEN X = 54:TA =
     136
430  XDRAW  NOT B + BB + 17 AT X1
      * 5,130
440  XDRAW B + BB + 17 AT X * 5,1
     30
450 VV =  PEEK ( - 16336)
460 X1 = X
470 B =  NOT B
480  IF TA = 149 AND BB = 2 THEN
     X = X + 1: GOTO 310
490  IF TA = 136 AND BB = 0 THEN
     X = X - 1: GOTO 310
500  IF TA = 149 AND BB = 0 THEN
      XDRAW  NOT B + 17 AT X * 5,
     130:BB = 2: GOTO 440
510  IF TA = 136 AND BB = 2 THEN
      XDRAW  NOT B + BB + 17 AT X
      * 5,130:BB = 0: GOTO 440
520  GOTO 310
600  REM  TASTI NUMERICI
610  IF X > 13 AND X < 17 AND TA =
     177 THEN Q1 = 1:Q2 = 70:Q3 =
     50:Q4 = 76: GOTO 810
620  IF X > 20 AND X < 24 AND TA =
     178 THEN Q1 = 3:Q2 = 103:Q3 =
     51:Q4 = 108: GOTO 810
630  IF X > 25 AND X < 29 AND TA =
     179 THEN Q1 = 5:Q2 = 131:Q3 =
     50:Q4 = 136: GOTO 810
640  IF X > 30 AND X < 34 AND TA =
     180 THEN Q1 = 7:Q2 = 156:Q3 =
     52:Q4 = 161: GOTO 810
650  IF X > 35 AND X < 39 AND TA =
```

Seguito listato Il campanaro.

```
     181 THEN Q1 = 9:Q2 = 180:Q3 =
     52:Q4 = 184: GOTO 810
660  IF X > 38 AND X < 43 AND TA =
     182 THEN Q1 = 11:Q2 = 200:Q3
      = 54:Q4 = 203: GOTO 810
670  IF X > 42 AND X < 46 AND TA =
     183 THEN Q1 = 13:Q2 = 218:Q3
      = 54:Q4 = 221: GOTO 810
680  IF X > 45 AND X < 49 AND TA =
     184 THEN Q1 = 15:Q2 = 234:Q3
      = 54:Q4 = 236: GOTO 810
690  IF X > 48 AND X < 52 AND TA =
     185 THEN Q1 = 249:Q2 = 244: GOTO
     730
700  IF X > 51 AND X < 55 AND TA =
     176 THEN Q1 = 260:Q2 = 255: GOTO
     730
710  POKE  - 16368,0
720  RETURN
730  XDRAW  NOT B + BB + 17 AT X1
      * 5,130: XDRAW 23 AT Q1,116
      : DRAW 22 AT Q2,52
740  IF (WA = 0 AND TA = 185) OR
     (WZ = 0 AND TA = 176) THEN 1
     440
750  IF WA AND TA = 185 THEN  XDRAW
     27 AT CG,CE:W = 0:WW = 0:WA =
     0:WD = 0: CALL 768,9,1,2,20
760  IF WZ AND TA = 176 THEN  XDRAW
     26 AT VB * 5,134:WZ = 0:V =
     0: CALL 768,10,1,1,5
770  FOR J = 1 TO 200: NEXT
780  XDRAW 22 AT Q2,52: XDRAW 23 AT
     Q1,116: DRAW  NOT B + BB + 1
     7 AT X1 * 5,130
790  POKE  - 16368,0
800  RETURN
810  XDRAW Q1 AT Q2,Q3
820  IF WD THEN 850
830  XDRAW  NOT B + BB + 17 AT X1
      * 5,130
840  XDRAW 23 AT Q4,116
850  DRAW Q1 + 1 AT Q2,Q3
860  CALL 768,(Q1 + 1) / 2,3,1,(Q
     1 * 2) + 9
870  IF WD THEN  RETURN
880 QQ =  INT (Q1 / 2 + 1)
890  FOR J = 1 TO 200: NEXT
900  GOSUB 1400
910 PU = PU + (9 - (Q1 + 1) / 2) *
     LI
925  VTAB 23: HTAB 26: PRINT PU
930  XDRAW Q1 + 1 AT Q2,Q3
940  IF WD THEN 960
950  XDRAW 23 AT Q4,130 - 14: XDRAW
      NOT B + BB + 17 AT X1 * 5,1
     30
960  DRAW Q1 AT Q2,Q3
970  POKE  - 16368,0
980  IF RI THEN RI = 0: POP : GOTO
     170
990  RETURN
1000  REM  FARFALLA
1010 WA = 1
1020  IF W THEN 1060
1030 W = 1:CF = 100:CH = 20:CG =
     20: XDRAW 27 AT 20,100
1040 CC =  INT (8 *  RND (1) + 1)

1050 CD = 78 + (CC = 2) * 31 + (C
     C = 3) * 59 + (CC = 4) * 84 +
     (CC = 5) * 108 + (CC = 6) *
     127 + (CC = 7) * 144 + (CC =
     8) * 160
1060 CG = CG + 2: IF CG >  = CD THEN
     X2 = X:X = CG / 5:TA = 176 +
     CC: GOTO 1110
1070 CE =  INT (15 *  RND (1) + 9
     0)
1080  XDRAW 27 AT CH,CF: XDRAW 28
      AT CG,CE: FOR JJ = 1 TO 10:
      NEXT : XDRAW 28 AT CG,CE: XDRAW
     27 AT CG,CE
1090 CF = CE:CH = CG
1100  RETURN
1110 WD = 1: GOSUB 600:DD = DD +
     1
1120  IF CC <  > N(DD) THEN  FOR
     J = 1 TO 1000: NEXT : GOTO 1
     440
1130  GOSUB 930:W = 0:WW = 0:WA =
     0:WD = 0:X = X2: XDRAW 27 AT
     CH,CF
1140  IF DD = NP THEN QQ = N(DD):
      GOTO 1420
1150  RETURN
1200  REM  TOPO
1210  IF V THEN 1230
1220 V = 1:VB = 5: XDRAW 26 AT 25
     ,134
1230 VA = VA + 1: IF VA < 2 THEN
      RETURN
1240 VA = 0
1250 VB = VB + 1
1260  XDRAW 26 AT (VB - 1) * 5,13
     4
1270  XDRAW 26 AT VB * 5,134
1280  IF VB = X OR VB = X - 1 OR
     VB = X - 2 THEN 1300
1290  RETURN
1300  XDRAW  NOT B + BB + 17 AT X
     1 * 5,130
```

Seguito listato Il campanaro.

```
1310  FOR JJ = 1 TO 3
1320  FOR J = 0 TO 48 STEP 16: ROT=
     J: XDRAW 17 AT X1 * 5,130: XDRAW
     17 AT X1 * 5,130: NEXT
1330  CALL 768,10,1,20,1
1340  NEXT
1350  XDRAW 17 AT X1 * 5,130: ROT=
     0
1360  GOTO 1440
1400  REM  CONTROLLO CAMPANA
1410 DD = DD + 1
1420  IF DD = NP AND QQ = N(DD) THEN
     DD = 0:RI = 1: RETURN
1430  IF QQ = N(DD) THEN  RETURN

1440  CALL 768,10,9,1,15
1450  HOME : VTAB 21: PRINT "CHE
     PECCATO... NON CE L'HAI FATT
     A A      COMPLETARE TUTTA LA
     SERIE DI MUSICHE DI PORFIRIO
      (MA QUANTE SARANNO?!?)."
1460  PRINT  CHR$ (4)"OPEN CAM.RE
     C"
1470  PRINT  CHR$ (4)"READ CAM.RE
     C"
1480  FOR J = 1 TO 10
1490  INPUT A$(J),A(J)
1500  NEXT
1510  FOR J = 1 TO 10
1520  IF A(J) < PU THEN 1550
1530  NEXT
1540  GOTO 1650
1550  FOR K = 9 TO J STEP  - 1
1560 A$(K + 1) = A$(K):A(K + 1) =
     A(K)
1570  NEXT
1580 A$(J) = NO$:A(J) = PU:J = 10

1590  PRINT  CHR$ (4)"DELETE CAM.
     REC"
1600  PRINT  CHR$ (4)"OPEN CAM.RE
     C"
1610  PRINT  CHR$ (4)"WRITE CAM.R
     EC"
1620  FOR J = 1 TO 10
1630  PRINT A$(J): PRINT A(J)
1640  NEXT
1650  PRINT  CHR$ (4)"CLOSE"
1660  TEXT : HOME : INVERSE : PRINT
     "..........TOP TEN - IL CAMPA
     NARO..........": NORMAL
1670  FOR J = 1 TO 10
1680  VTAB J + 3: PRINT A$(J); TAB(
     25);A(J)
1690  NEXT
1700  POKE  - 16368,0: VTAB 20: PRINT
     "PREMI -RETURN- PER RIPARTIR
     E": PRINT "      UN ALTRO TA
     STO PER FINIRE";: GET R$: PRINT
     R$: IF R$ =  CHR$ (13) THEN
      RUN 50
1710  END
2000  DATA  1,70,50,3,103,51,5,13
     1,50,7,156,52,9,180,52,11,20
     0,54,13,218,54,15,234,54,21,
     249,55,21,260,55
2010  DATA  24,20,99,25,20,131,29
     ,70,32,30,103,32,31,131,32,3
     2,156,32,33,180,32,34,200,32
     ,35,218,32,36,234,32,37,249,
     32,38,260,32
2020  DATA  76,108,136,161,184,20
     3,221,236,249,260
```

Sviluppo colonnare condizionato

*L'autore dell'articolo Vincenzo Carnemolla ci invia alcuni aggiornamenti sul programma in questione (***Personal Software** *n. 33).*

L'appunto che mi viene fatto più spesso è che finito l'input, lo schermo rimane bianco per un tempo che, a seconda del condizionamento, può essere piuttosto lungo, e quindi non si ha la certezza che il programma giri regolarmente. Per ovviare a questo inconveniente ho pensato di aggiungere le seguenti linee:

```
500 W2=W2+1
502 PRINT TH$X$ (0) Y$ (20); W2
504 GOTO 50
```

in questo modo in basso a sinistra dello schermo comparirà un numero che indica la colonna presa in esame in quel momento. Quindi dal progredire di questo numero si ha la certezza del buon funzionamento del programma stesso.

Alla fine dello sviluppo, questo valore indica il numero delle colonne integrali del sistema che si è condizionato.

Lucy writer per Spectrum

Un potente word processor di facile utilizzo, particolarmente mirato ai possessori di microdrive

di Maurizio Rogante e Andrea Nordio

Presentazione

Lucy writer, come facilmente intuibile, è un potente wp per lo ZX Spectrum. Sfrutta al meglio le caratteristiche del calcolatore, soprattutto per chi possiede anche i microdrive. È di facile uso, essendo interattivo.

Il programma, nonostante sia scritto in Basic, consta di alcune routine in linguaggio macchina, che consentono di aumentarne la velocità nei punti dove il Basic risulterebbe troppo lento. Il programma, in fase di stampa, giustifica a destra e a sinistra il testo con un numero a scelta di caratteri per linea, provvedendo anche a dividere in sillabe le parole interrotte a fine riga per una perfetta impaginazione.

Se per qualsiasi motivo si dovesse interrompere il programma, sarà sufficiente battere "Goto Restart" ed Enter per farlo ripartire.

Struttura del programma

Il testo è contenuto in un'unica stringa ed è mostrato sul video, a pagine di 21 righe per 32 colonne; le pagine sono 35 per un totale di 23.520 caratteri, più che sufficienti per evadere la corrispondenza o per redigere brevi dattiloscritti.

Il testo non è video-orientato, cioè quello che appare sul video non coincide con ciò che verrà stampato: questa non è una limitazione perché non ha nessun interesse sapere come lo scritto sarà stampato, infatti sarà sempre conforme ai nostri voleri.

Appena lanciato, passata la presentazione, si entra in ambiente scrittura, e appare la prima pagina; sulla prima riga compare la scritta in inverse indicante l'inizio del testo: in basso, sotto una linea di separazione, è indicato il numero della pagina su cui ci si trova, (e che è la pagina corrente). Ora potete iniziare a scrivere dal punto ove è posizionato il cursore di scrittura ("-") senza curarvi di andare a capo correttamente, ma scrivendo tutto di seguito come fosse tutta un'unica riga badando a inserire lo spazio all'inizio della riga successiva quando la parola precedente terminava esattamente a fine riga.

Quando si vuole andare a capolinea si prema Enter: comparirà una nota musicale; quest'ultima avvertirà il calcolatore in fase di stampa che il paragrafo è finito. Se per errore la si cancella va riscritta. Resta attivo il tasto Delete (Cap Shift +0). Tutti i tasti hanno il repeat.

Premendo Simbol Shift +e (>=) si aziona il tabulatore che sposta il cursore di cinque colonne per default. Premendo S.S.+h (↑) si aziona il modo calcolatore: verrà richiesta una formula, il cui risultato verrà stampato sul video dove era il cursore, e inserito nel testo.

Premendo S.S.+a (Stop) si passa in ambiente comandi: il loro elenco e relative spiegazioni compare in tabella 1.

Dall'ambiente scrittura, premendo Edit (C.S.+1), si entra in ambiente editor. Ora è comparso il cursore di edit (in inverse) ed è possibile posizionarlo altrove, usando i tasti freccia, per eseguire correzioni; per sostituire la lettera del cursore basta premere il tasto corrispondente alla nuova lettera. Per più complesse modifiche bisogna ripremere Edit: si entra così in ambiente alterazioni. Qui con comandi a tasto unico (elencati in tabella 2) si possono apportare le modifiche al testo.

Da questo ambiente, premendo Stop (S.S.+a), si ritorna in ambiente editor; premendolo nuovamente si ripassa in ambiente scrittura con la possibilità di far tornare il cursore dov'era originariamente oppure di lasciarlo dove si trovava il cursore dell'ambiente editor.

Nelle pagine successive a quella iniziale, la prima riga conterrà l'ultima riga della pagina precedente: ciò per facilitare la stesura del testo.

Spiegazione di alcuni comandi

Change - Cambia una stringa con un'altra. La ricerca è effettuata a cominciare dal primo carattere della pagina corrente, ed è possibile per stringhe fino a 20 caratteri. La stringa individuata è segnalata dallo spostamento del cursore. Ora premendo "c" avviene la sostituzione; premendo "a" la ricerca prosegue; premendo Edit (C.S.+1) si passa in ambiente editor.

Locate - Ricerca di una stringa. Il funzionamento è simile al Change salvo che il tasto "c" non ha alcun effetto.

Print - Stampa su carta il testo perfettamente giustificato con il C.P.L. voluto, interrompendosi quando si preme S.S.+a (Stop) o si incontra il marcatore "$★$".

Alla fine del testo è infatti necessario porre il suddetto marcatore a inizio riga, altrimenti la stampante continuerà a scrivere spazi fino all'esaurimento delle pagine. Ponendo, a inizio riga, il marcatore ":" la stringa immediatamente do-

Tabella 1 - Elenco dei comandi.

aid	Pone i numeri di riga.
back	Indietro di una pagina.
bot	Va all'ultima pagina.
change	Cambia una parola con un'altra, a partire dalla pagina corrente.
colour	Per la scelta dei colori di Ink e Paper.
cpl n	Setta a n i caratteri per linea della stampa (default 32).
delpage	Pulisce la pagina corrente.
edit	Passa direttamente in ambiente Edit.
forw	Va avanti di una pagina.
home	Porta il cursore scrittura in alto a sinistra.
go n	Va alla pagina n e la rende corrente.
help	Mostra l'elenco dei comandi e delle funzioni di alterazione.
inform	Informazioni generali.
locate	Trova, a partire dalla pagina corrente, una parola data.
new	Comincia tutto da capo.
print	Stampa su stampante.
ret	Torna in ambiente scrittura.
sylls	Scomposizione sillabica di una parola o frase della lingua italiana.
tab n	Setta il tabulatore perché salti ogni n colonne.
top	Va alla prima pagina.
type	Stampa su stampante senza giustificare a 32 colonne (stampa di servizio).
view	Mostra tutto il testo sul video con lo scroll, premendo un tasto qualsiasi. Premendo Stop (S.S. + a) si torna ai comandi.

Comandi per la gestione della memoria di massa

save	Salvataggio
load	Caricamento con sistemazione a partire dalla prima pagina.
append	Caricamento con sistemazione a partire da una pagina specificata.

I comandi di cui sopra, se seguiti da un asterisco si riferiscono ai microdrive.

format	Per la formattazione di una nuova cartuccia, a cui viene dato il nome "Testi".
cat n	Catalogo della cartuccia nel microdrive n.
erase	Cancellazione di un file dati nel microdrive.

N.B. - Tutti i comandi in cui compare la lettera n (tipo go n) vogliono un argomento, che dev'essere intero positivo e distare almeno uno spazio dalla parola precedente.

Tabella 2 - Elenco funzioni di alterazione.

a	Aggiungere uno spazio del cursore spostando solo la linea corrente.
c	Cancella la linea corrente dal cursore in poi.
d	Cancella uno o più caratteri dopo il cursore spostando tutto il testo.
D	Cancella un carattere dopo il cursore spostando solo la linea corrente (è il duale di "a").
e	Cancella una o più righe dalla corrente verso il basso spostando in su tutto il testo.
l	Aggiunge una o più linee sopra la linea corrente, spostando in giù tutto il testo. Le ultime linee dell'ultima pagina vanno così perse.
i	Inserisce una stringa di caratteri prima del cursore.
m	Muove blocchi di linee anche da una pagina all'altra.
n	Sposta a capo la linea corrente dal cursore alla fine.
N	Sposta la linea a capo della corrente dopo il cursore. Eventuali caratteri in eccesso saranno perduti.
Stop	Torna in ambiente Edit (S.S. + a).

N.B. - I comandi: Change, Locate, Print e la funzione "d" sono spiegati più diffusamente nel testo.
I comandi e le funzioni vanno battuti in minuscolo. Le funzioni in maiuscolo vanno date premendo anche Caps Shift.

Figura 1 - Hardcopy del programma.

```
     Si racconta che il principe
di  Conde'  dormi' profondamente
la  notte  avanti la giornata di
Rocroi:  ma, in primo luogo, era
molto affaticato; secondariamen-
ta  aveva gia' date tutte le di-
sposizioni  necessarie, e stabi-
lito  cio'  che dovesse fare, la
mattina.

da "I Promessi Sposi" Cap. II


** INIZIO DEL TESTO **
     Si racconta che il principe
 di Conde' dormi' profondamente
la notte avanti la giornata di R
ocroi: ma, in primo luogo, era m
olto affaticato; secondariamente
 aveva gia' date tutte le dispos
izioni necessarie, e stabilito c
io' che dovesse fare, la mattina
.♪
♪
da "I Promessi Sposi" Cap. II♪
$*$♪
```

po di esso verrà centrata nella riga; se l'ultimo carattere è la nota, quest'ultima verrà ignorata.

L'opzione definita dal ":" ha effetto solo se utilizzata all'inizio di un capoverso.

Funzione "d" - Delete. Premendo il tasto "d" in ambiente alterazioni, si aziona la funzione delete che cancella dal testo gruppi di caratteri. Appare un cursore lampeggiante dov'era il cursore di editor, spostabile coi tasti freccia (senza C.S.!). Premendo Stop si rinuncia alla funzione. I caratteri cancellati saranno quelli in inverse più il cursore. Se si tenta di cancellare oltre la fine del paragrafo, i caratteri in eccesso saranno ignorati. Non si può oltrepassare la pagina corrente. Premere Enter per far avviare la modifica.

Modifiche al programma

La subroutine che effettua materialmente la stampa su carta si trova in linea 3500, ed è predisposta per una stampante Spectrum dedicata. Per renderla compatibile alla vostra stampante (non Spectrum-dedicata) dovrete modificarla opportunamente. Ricordatevi di mettere il Return al termine della subroutine.

Chi volesse aggiungere nuovi comandi o funzioni di alterazione, sappia che la routine e la tabella dei primi inizia in linea 400, mentre la routine e la tabella delle seconde inizia a 1100.

Principali routine

500 - Preparazione alla giustificazione.
600 - Ricerca dei paragrafi nel testo per la stampa.
700 - Stampa di servizio (comando type).
800 - Scomposizione sillabica.
1142 - Cancellazione dei caratteri dal testo.
2000 - Presentazione iniziale e sistemazione del linguaggio macchina.
2034 - Inserimento di stringhe nel testo.
3000 - Giustificazione.
8000 - Creazione delle costanti e dell'area testo.
8530 - Passaggio parametri al linguaggio macchina.

Principali variabili

R,C - Coordinate cursore scrittura.
R1,C1 - Coordinate cursore Edit.
TAB - Parametro di tabulazione (default cinque).
CPL - Caratteri per linea (default 32).
CC - Spazi da inserire per la giustificazione.
AA,BB - Primo e ultimo carattere nel testo per operazioni varie.
POS - Posizione nel testo del cursore di Edit.
S (20) - Array contenente il numero d'ordine dell'ultima lettera di ogni sillaba della parola esaminata. S(1) è sempre zero. Massimo 19 sillabe.
R$ - Contiene il comando o la stringa da manipolare nel corso della stampa.
F$ - Contiene il capoverso da trattare.
N$ - Nome del testo.
T$(23520) - Il testo.

Routine in linguaggio macchina

65447 - Trova una stringa nel testo, memorizzata da 65427, e lunga al massimo 20 caratteri.
65489 - Trova la prima nota dal cursore.
65504 - Pulisce tutto il testo.
65524 - Sposta su o giù parte del testo. ■

Listato 1 - Il programma Lucy writer.

```
  10 REM by M.Rogante & A.Nordio
  11 REM
  12 GO SUB 8e3: GO SUB 2e3: PAU
SE al: BORDER al: INK 4: POKE 23
624,4: CLS : GO SUB pg: LET c=al
  15 LET r=si
  20 PRINT AT r,c;" "
  25 PAUSE al: LET k=CODE INKEY$
  BEEP .01,45
  35 IF k<32 OR k>164 OR k=94 TH
EN GO TO 85
  40 PRINT AT r,c;CHR$ k: LET t$
(pc+r*mo+c+si)=CHR$ k
  45 LET c=c+1: IF c>31 THEN GO
TO 60
  46 GO TO 20
  60 LET c=c-mo: LET r=r+1
  65 IF r<22 THEN GO TO 20
  75 GO TO 1000
  85 IF k=13 THEN GO TO 130
  90 IF k=12 THEN GO TO 145
 100 IF k=7 THEN GO TO 245
 105 IF k=94 THEN GO TO 180
 110 IF k=226 THEN GO TO 400
 115 IF k=200 THEN GO TO 1410
 120 GO TO 25
 130 PRINT AT r,c;"♪"
 135 LET t$(pc+r*mo+c+si)="♪": L
ET c=mo: GO TO 50
```

```
 147 PRINT AT r,c;" "
 150 LET c=c-si
 155 IF c>=al THEN GO TO 175
 160 LET c=31: LET r=r-si
 165 IF r>=si THEN GO TO 175
 170 LET r=si: LET c=al: GO TO 2
0
 175 LET t$(pc+r*mo+c+si)=" ": G
O TO 20
 180 INPUT "Formula? "; LINE a$:
 GO SUB pg: IF a$="" THEN GO TO
25
 190 LET a$=STR$ VAL a$
 200 IF r=21 AND c+LEN a$>mo31 A
ND pa=mxpg THEN GO TO 1025
 210 LET t$(pc+r*mo+c+si TO pc+r
*mo+c+LEN a$)=a$
 215 GO SUB mp: LET c=c+LEN a$:
GO TO 45
 245 PRINT INVERSE si;AT r,c;" "
 250 LET r1=r: LET c1=c
 260 IF INKEY$<>"" THEN GO TO 26
0
 265 LET c$=t$(pc+r1*mo+c1+si)
 275 PRINT AT r1,c1; INVERSE si;
c$: PAUSE al
 280 LET k=CODE INKEY$
 285 IF k<32 OR k>164 THEN GO TO
 300
 290 PRINT AT r1,c1;CHR$ k: LET
t$(pc+r1*mo+c1+si)=CHR$ k: GO TO
 285
```

Seguito listato Lucy writer.

```
 300 PRINT AT r1,c1;c$
 301 IF k=226 THEN GO TO 360
 305 IF k=7 THEN GO TO 1100
 310 LET r1=r1+(k=10)-(k=11): LE
T c1=c1-(k=8)+(k=9)
 312 IF r1>21 THEN LET r1=si
 314 IF r1<si THEN LET r1=21
 316 IF c1>31 THEN LET c1=al
 318 IF c1<al THEN LET c1=31
 320 GO TO 265
 360 PRINT #si;AT si,al;"QUI ?
": PAUSE al
 365 IF INKEY$="n" THEN GO TO 37
2
 370 LET r=r1: LET c=c1
 372 GO SUB pg: IF r>21 THEN GO
TO 1025
 374 GO TO 20
 400 PRINT AT r,c;t$(pc+r*mo+c+s
i)
 402 GO SUB pg: INPUT FLASH si;"
COMANDI"; FLASH al;"  ===>"; LIN
E r$
 404 GO SUB pg: LET k=al: LET ar
g=-si: IF LEN r$<3 THEN GO TO 46
6
 406 FOR i=si TO LEN r$: IF r$(i
)=" " THEN LET k=i: GO TO 410
 408 NEXT i
 410 IF k>al THEN LET arg=VAL r$
(k+si TO ): LET r$=r$( TO k-si)
 418 IF r$="forw" THEN GO TO 105
0
 420 IF r$="back" THEN GO TO 106
6
 422 IF r$="save" OR r$="save*"
THEN GO TO 5000
 424 IF r$="load" OR r$="load*"
THEN GO TO 5006
 426 IF r$="inform" THEN GO TO 7
070
 428 IF r$="edit" THEN GO TO 102
0
 429 IF r$="colour" THEN GO TO 5
050
 430 IF r$="ret" THEN GO TO 20
 432 IF r$="go" THEN GO TO 4100
 434 IF r$="top" THEN GO TO 950
 436 IF r$="bot" THEN GO TO 956
 438 IF r$="delpage" THEN GO TO
4110
 440 IF r$="view" THEN GO TO 145
0
 442 IF r$="tab" THEN GO TO 1420
 444 IF r$="change" THEN GO TO 4
205
 446 IF r$="locate" THEN GO TO 4
200
 448 IF r$="sylls" THEN GO TO 15
10
 450 IF r$="home" THEN GO TO 150
0
 451 IF r$="cat" THEN GO TO 5034
 452 IF r$="append" OR r$="appen
d*" THEN GO TO 5012
 453 IF r$="format" THEN GO TO 5
032
 454 IF r$="new" THEN RUN
 456 IF r$="print" OR r$="type"
THEN GO TO 600
 458 IF r$="cpl" THEN GO TO 720
 460 IF r$="aid" THEN GO TO 690
 462 IF r$="erase" THEN GO TO 50
38
 464 IF r$="help" THEN GO TO 120
0
 466 PRINT #si;AT si,al;w$;AT si
,al;"NON ESISTE """;r$;"""": BEE
P 1,0: PAUSE 60: GO TO cmd
 468 PRINT #si;AT si,al;"ARGOMEN
TO ERRATO": BEEP 1,0: PAUSE 60:
GO TO cmd
 500 LET i=si: IF i>=brk THEN LE
T p$=f$: GO SUB pt: RETURN
 505 LET a$=f$(i TO i+cpl-si)
 510 LET r$=a$( TO cpl): LET s$=
a$(cpl+si TO ): LET j=i+cpl
 512 IF INKEY$=" STOP " THEN RET
URN
 514 LET b$=s$+f$(j TO j+cpl-si)
 518 IF r$(cpl)=" " OR b$(si)="
" THEN GO TO 552
 520 FOR k=cpl-si TO si STEP -si
: IF r$(k)<>" " THEN NEXT k
 525 FOR w=pr TO cpl: IF b$(w)<>
" " THEN NEXT w
 530 LET j$=r$(k+si TO )+b$( TO
w-si): GO SUB 800
 535 FOR y=si TO 20: IF s(y)<=cp
l-si-k THEN NEXT y
 540 IF s(y-si)=al THEN GO TO 55
0
 545 LET r$(k+si TO )=j$( TO s(y
-si))+"-": LET b$=j$(s(y-si)+si
TO )+b$(w TO ): GO TO 552
 550 LET r$(k+si TO )=e$: LET b$
=j$+b$(w TO )
 552 GO SUB 3e3: LET i=j: IF i>=
brk THEN GO TO 580
 554 LET a$=b$: IF LEN a$<cpl TH
EN LET a$=a$+e$( TO cpl-LEN a$)
 556 GO TO 510
 580 IF LEN b$>cpl THEN GO TO 59
0
 585 LET r$=b$: GO SUB 3078: RET
URN
 590 LET r$=b$( TO cpl): LET b$=
b$(cpl+si TO )
 592 IF b$=e$( TO LEN b$) THEN G
O SUB 3078: RETURN
 594 GO TO 518
 600 GO SUB qs: IF r$="type" THE
N GO TO 700
 604 GO TO 608+22*(t$(aa)=":")+4
46*(t$(aa TO aa+pr)="$*$")
 610 GO SUB 1600
 616 LET f$=t$(aa TO i+j-pr): LE
T f$=f$+e$( TO cpl-LEN f$+cpl*(I
NT (LEN f$/cpl)))
 618 LET brk=LEN f$-cpl+si: LET
aa=i+mo: IF aa>bb THEN GO TO 105
4
 620 GO SUB 500: IF INKEY$=" STO
P " THEN GO TO restart
 622 GO TO 604
 630 LET r$=t$(aa+si TO aa+mo31)
+" ": LET aa=aa+mo
 632 FOR i=cpl TO pr STEP -si: I
F r$(i)<>" " THEN GO TO 636
 634 NEXT i: GO TO 604
 636 IF r$(i)="♪" THEN LET i=i-s
i
 638 LET r$=w$( TO (cpl-LEN r$(
TO i))/pr)+r$( TO i): LET p$=r$:
 GO SUB pt: GO TO 604
 690 GO SUB 2100: PAUSE al: GO T
O 1054
 700 FOR i=aa TO bb STEP mo: LET
 p$=t$(i TO i+mo31): GO SUB pt
 702 IF INKEY$=" STOP " OR r$( T
O 3)="$*$" THEN GO TO restart
 704 NEXT i: GO TO restart
 720 IF arg<1 THEN GO TO 468
 722 LET cpl=arg: DIM e$(cpl+si)
: GO SUB pg: GO TO cmd
 750 LET t$(pos-c1 TO pc+r1*mo+m
o)=t$(pos-c1 TO pos-si)+" "+t$(p
os TO pc+r1*mo+mo31): GO TO 1180
 754 LET t$(pos TO pos-c1+mo31)=
t$(pos+si TO pos-c1+mo31): GO TO
 1180
 770 INPUT "Quante linee? ";l: L
ET fa=(k$="l")
 774 IF fa THEN GO SUB 1187: GO
TO 1180
 776 GO SUB 1154: GO TO 1180
 800 LET k$=j$: DIM s(20): LET p
p=pr: LET ls=LEN j$
 805 IF k$(LEN k$)<CHR$ mo OR k$
(LEN k$)>CHR$ 164 THEN LET k$=k$
( TO LEN k$-si): GO TO 805
 810 LET h=LEN k$
 812 IF h<=2 THEN GO TO 872
 814 LET ii=si
 816 GO SUB 876
```

Seguito listato Lucy writer.

```
 818 IF ll=0 THEN GO TO 866
 820 LET ll1=ll
 822 LET ii=ii+si
 824 IF ii>h THEN GO TO 872
 826 IF k$(ii)="h" OR k$(ii)="H"
THEN GO TO 822
 828 GO SUB 876
 830 IF ll=al THEN GO TO 848
 832 IF ll1=2 OR ll=2 THEN GO TO
 840
 834 LET jj=ii-si
 836 GO SUB 896
 838 GO TO 810
 840 LET ii=ii+1
 842 IF ii>h THEN GO TO 872
 844 LET ll1=ll
 846 GO TO 828
 848 LET ii=ii+1
 850 IF ii>h THEN GO TO 872
 852 GO SUB 876
 854 IF ll=al THEN GO TO 860
 856 LET jj=ii-pr
 858 GO TO 836
 860 GO SUB 904
 862 IF s=0 THEN GO TO 834
 864 GO TO 856
 866 LET ii=ii+1
 868 IF ii>h THEN GO TO 872
 870 GO TO 816
 872 LET s(pp)=ls
 874 RETURN
 878 LET l$=k$(ii)
 880 IF l$="i" OR l$="I" OR l$="
u" OR l$="U" THEN GO TO 892
 882 IF l$="a" OR l$="A" OR l$="
e" OR l$="E" OR l$="o" OR l$="O"
 THEN GO TO 888
 884 LET ll=al
 886 RETURN
 888 LET ll=si
 890 RETURN
 892 LET ll=pr
 894 RETURN
 898 LET s(pp)=jj+s(pp-si)
 900 LET pp=pp+si: LET k$=k$(jj+
si TO )
 902 RETURN
 906 LET l$=k$(ii-si)
 908 LET m$=k$(ii)
 910 IF l$=m$ THEN GO TO 924
 912 IF l$="g" AND m$="n" THEN G
O TO 920
 914 IF l$="l" OR l$="m" OR l$="
n" OR l$="r" OR l$="v" OR l$="z"
 THEN GO TO 924
 918 IF l$="s" OR m$="r" THEN GO
 TO 928
 920 IF l$="t" THEN GO TO 924
 922 IF m$="h" OR m$="l" THEN GO
 TO 928
 924 LET s=al
 926 RETURN
 928 LET s=si
 930 RETURN
 950 LET pa=si: LET pc=-mo: GO T
O 1054
 956 LET pa=mxpg: LET pc=pa*nc-n
c-mo: GO TO 1054
1000 IF pa=mxpg THEN GO TO 1025
1005 CLS : LET pa=pa+si: LET pc=
pc+nc
1010 PRINT AT al,al;t$(pc+si TO
pc+mo);AT si,al;t$(pc+mo33 TO pc
+nc+mo): GO SUB pg: GO TO 15
1020 LET r=21: LET r1=r: LET c1=
al: GO TO 265
1025 PRINT #si;AT si,al;w$;AT si
,al;"NON C'E' PIU' SPAZIO!": BEE
P si,al: PAUSE 60: GO SUB pg: GO
 TO 1020
1050 IF pa=mxpg THEN GO TO cmd
1052 LET pa=pa+si: LET pc=pc+nc
1054 GO SUB mp: GO TO cmd
1066 IF pa=si THEN GO TO cmd
1068 LET pa=pa-si: LET pc=pc-nc:
 GO TO 1054
1100 PRINT AT r1,c1; INVERSE si;
c$;#si;AT si,al;w$;AT si,al;"MOD
O ALTERAZIONI": LET pos=pc+r1*mo
+c1+si
1102 PAUSE al: LET k$=INKEY$
1104 IF k$="c" THEN GO TO 1130
1105 IF k$="i" THEN GO TO 2020
1106 IF k$="l" THEN GO TO 770
1107 IF k$="e" THEN GO TO 770
1108 IF k$="m" THEN GO TO 2200
1109 IF k$="d" THEN GO TO 2070
1110 IF k$="D" THEN GO TO 754
1111 IF k$="n" THEN GO TO 2250
1112 IF k$="a" THEN GO TO 750
1113 IF k$=" STOP " THEN GO TO 1
140
1114 IF k$="N" THEN GO TO 2258
1120 GO TO 1102
1130 LET t$(pos TO pc+r1*mo+mo)=
e$: GO TO 1180
1140 PRINT AT r1,c1;t$(pos): GO
SUB pg: GO TO 265
1142 LET aa=pos-c1: LET bb=mxca:
 GO SUB 1600: LET a=6+PEEK 23627
+256*PEEK 23628: POKE 65534,176:
 IF i+j-pos<n THEN LET n=i+j-pos
1143 IF i+j>mxca THEN LET v1=pos
+n+a-si: LET v2=pos+a-si: LET v3
=mxca-v1+a: GO TO 1152
1144 LET f$=t$(pos+n TO i+mo31):
 LET t$(i+mo31-n+si TO i+mo31)=w
$: LET t$(pos TO i+mo31-n)=f$
1145 IF j>n THEN RETURN
1146 LET k=INT ((ABS (j-n)+mo)/m
o): LET v1=i+mo+a-si: LET v2=i+m
o-mo*k+a-si: LET v3=mxca-i-mo
1148 GO SUB 8530: RANDOMIZE USR
65524
1150 LET t$(mxca-mo*k TO )=w$: R
ETURN
1152 IF v3>al THEN GO SUB 8530:
RANDOMIZE USR 65524: LET t$(mxca
-n+si TO )=w$: RETURN
1153 LET t$(pos TO )=w$: RETURN
1154 IF mxca-(pc+r1*si+mo)<l*mo
THEN RETURN
1156 LET pos1=pos: LET pos=pos-c
1: LET c2=c1: LET c1=al: LET n=m
o*l: GO SUB 1142: LET pos=pos1:
LET c1=c2: RETURN
1160 GO SUB pg: IF pa>si THEN PR
INT AT al,al;t$(pc+si TO pc+mo)
1162 PRINT AT si,al;t$(pc+mo33 T
O pc+nc+mo): RETURN
1180 GO SUB mp
1182 LET c$=t$(pos): GO TO 1100
1188 IF l*mo>mxca-pos+c1 THEN LE
T t$(pos-c1 TO )=w$: RETURN
1190 LET pos1=pos: LET pos=pos-c
1: LET c2=c1: LET c1=al: LET a$=
"": FOR z=si TO l: LET a$=a$+w$:
 NEXT z: GO SUB 2034: LET pos=po
s1: LET c1=c2: RETURN
1200 CLS : PRINT INVERSE si; FLA
SH si;"MODO COMANDI"
1210 PRINT '"forw","back","infor
m","edit","ret","go n","top","bo
t","delpage","view","tab","chang
e","locate","sylls","home","new"
,"print","type","cpl","aid","hel
p",
1215 PRINT "save","save*","load"
,"load*","append","append*","for
mat","cat","erase"
1220 PRINT ' INVERSE si; FLASH s
i;"MODO ALTERAZIONI"
1230 PRINT '"c  ,  i  ,  l  ,  e
  ,  m  ,  d D  ,  n  ;  N  ;  a
 , STOP "
1306 PAUSE al: GO TO 1054
1410 PRINT AT r,c;t$(pc+r*mo+c+s
i): LET c=tab*(INT (c/tab)+1)
1412 IF c>31 THEN LET c=al
1414 GO TO 20
1420 IF arg>mo31 OR arg<si THEN
GO TO 468
1422 LET tab=arg: GO SUB pg: GO
TO cmd
```

Seguito listato Lucy writer.

```
1450 CLS : PRINT AT 21,al;: POKE
 23692,255
1455 FOR i=si TO mxca STEP mo
1460 PRINT t$(i TO i+mo31)
1462 IF INKEY$="" THEN GO TO 146
2
1464 IF INKEY$=" STOP " THEN GO
TO restart
1465 NEXT i
1470 PRINT #si;"Premi un tasto":
 PAUSE al: GO TO restart
1500 LET r=si: LET c=al: GO TO c
md
1510 CLS : PRINT INVERSE 1;"
SCOMPOSIZIONE  SILLABICA     ": P
LOT al,166: DRAW 255,al: PLOT al
,164: DRAW 255,al
1512 INPUT "PAROLA O FRASE DA SI
LLABARE ",,,j$: PRINT AT 3,al;"S
OLUZIONE:",,,,
1514 GO SUB 800
1520 FOR i=pr TO 20
1525 IF s(i)<>LEN j$ THEN PRINT
j$(s(i-si)+si TO s(i));"-";: NEX
T i
1530 PRINT j$(s(i-si)+si TO ): P
AUSE al: GO TO 1054
1600 LET dd=aa+5: POKE 65494,INT
 (dd/256): POKE 65493,dd-256*PEE
K 65494
1602 LET v1=si+bb-aa: POKE 65498
,INT (v1/256): POKE 65497,v1-256
*PEEK 65498
1604 LET v2=aa+v1-USR 65489-pr:
LET i=(INT (v2/mo)*mo)+si: LET j
=v2-i+pr
1606 IF i>mxca THEN LET i=mxca-m
o31: LET j=mo: RETURN
1608 RETURN
2002 FOR i=65447 TO 65535: READ
a$: POKE i,VAL a$: NEXT i
2010 BORDER 5: PAPER al: CLS : I
NK 5: PLOT 56,72: DRAW OVER si;9
0,90,243: INK 3: FLASH si: PRINT
 BRIGHT si;AT 9,14;"LUCY";AT 11,
13; INVERSE si;"WRITER": FLASH a
l
2012 RETURN
2020 INPUT a$: IF a$=" STOP " OR
 LEN a$>mxca-pos THEN GO TO 1180
2022 GO SUB 2034: GO TO 1180
2034 LET aa=pos-c1: LET bb=mxca
2036 LET a=5+PEEK 23627+256*PEEK
 23628: POKE 65534,184: GO SUB 1
600: IF i+j+LEN a$>mxca THEN GO
TO 2060
2038 LET k=INT ((LEN a$+j-si)/mo
): IF LEN a$<mo-j THEN GO TO 204
8
2040 LET v1=mxca-mo*k+a: LET v2=
mxca+a: LET v3=v1-a-i-mo+si
2046 IF v3>al THEN GO SUB 8530:
RANDOMIZE USR 65524
2048 LET f$=t$(pos TO i+j-si): L
ET t$(pos TO i+mo*k+mo31)=w$: LE
T t$(pos TO pos+LEN a$)=a$
2050 LET t$(pos+LEN a$ TO pos+LE
N a$+LEN f$)=f$: RETURN
2060 LET v2=mxca+a: LET v1=v2-LE
N a$: LET v3=si+v1-a-pos
2062 GO SUB 8530: RANDOMIZE USR
65524
2064 LET t$(pos TO pos+LEN a$-si
)=a$: RETURN
2070 PRINT #si;AT si,al;w$;AT si
,al;"DELETING": LET r2=r1: LET c
2=c1: LET pos1=pos
2072 PRINT AT r2,c2; FLASH NOT C
ODE INKEY$; INVERSE si;t$(pos1):
 LET k=CODE INKEY$
2074 IF k=13 THEN LET n=pos1-pos
+si: GO SUB 1142: GO TO 1180
2076 IF k=226 THEN GO TO 1180
2078 IF k=56 AND c2<mo31 THEN PR
INT AT r2,c2; INVERSE si;t$(pos1
): LET c2=c2+si: LET pos1=pos1+s
i: GO TO 2072
2080 IF k=53 AND c2>al AND NOT (
r2=r1 AND c2<=c1) THEN PRINT AT
r2,c2;t$(pos1): LET c2=c2-si: LE
T pos1=pos1-si: GO TO 2072
2082 IF k=54 AND r2<21 THEN PRIN
```

ZX SPECTRUM

```
T AT r2,c2; INVERSE si;t$(pos1 T
O pos1+mo): LET r2=r2+si: LET po
s1=pos1+mo: GO TO 2072
2084 IF k=55 AND r2>r1 AND NOT (
r2=r1+si AND c2<c1) THEN LET r2=
r2-si: LET pos1=pos1-mo: PRINT A
T r2,c2;t$(pos1 TO pos1+mo)
2086 GO TO 2072
2100 FOR i=si TO 21: PRINT AT i,
al; INVERSE si;" " AND i<10;i; I
NVERSE al;" ";t$(pc+i*mo+si TO p
c+i*mo+mo31-pr)
2102 NEXT i: RETURN
2200 PRINT #si;AT si,al;"Inserim
ento parametri": GO SUB 2100
2205 INPUT "Da quale pagina? ";p
1: INPUT "A quale pagina? ";p2
2210 INPUT "Da quale riga? ";l1:
 INPUT "A quale riga? ";q1: INPU
T "In quale riga? ";l2: LET q1=q
1-l1+si
2220 IF p1>mxpg OR p1<si OR l1<s
i OR p2>mxpg OR l2<si OR p2<si O
R l1>21 OR q1<si OR q1>21 OR l2>
21 OR 22-l2<q1 THEN PRINT #si;AT
 si,al;"PARAMETRI ERRATI": BEEP
si,70: PAUSE 60: GO TO 2205
2225 LET t$(p2*nc-nc+si+(l2-si)*
mo TO p2*nc-nc+(l2-si)*mo+q1*mo)
=t$(p1*nc-nc+si+(l1-si)*mo TO p1
*nc-nc+(l1-si)*mo+q1*mo): GO TO
1180
2250 LET t$(pc+(r1+si)*mo+si TO
pc+(r1+si)*mo+mo)=t$(pos TO pc+r
1*mo+mo)
2252 LET t$(pos TO pc+r1*mo+mo)=
e$: GO TO 1180
2258 IF mxpg=pa AND r1=22 THEN G
O TO 1180
2260 LET t$(pos TO pc+r1*mo+mo)=
t$(pc+(r1+si)*mo+si TO pc+(r1+si
)*mo+mo): LET t$(pc+(r1+si)*mo+s
i TO pc+(r1+si)*mo+mo)=w$: GO TO
 1180
3000 FOR k=cpl TO si STEP -si
3015 IF r$(k)=" " THEN NEXT k: G
O TO 3064
3025 LET cc=cpl-k: LET cc1=cc: L
ET r$=r$( TO k)
3030 FOR n=si TO LEN r$: IF r$(n
)=" " THEN NEXT n
3032 IF n=pr THEN LET r$=r$(pr T
O ): LET cc=cc+si: LET cc1=cc
3034 IF cc=al THEN GO TO 3064
3036 FOR w=n TO LEN r$
3037 IF r$(w)=" " THEN GO TO 304
8
3038 NEXT w
3040 IF cc=cc1 THEN GO TO 3064
3042 IF cc>al THEN GO TO 3036
3044 GO TO 3064
3048 LET r$=r$( TO w)+" "+r$(w+s
i TO ): LET cc=cc-si: IF cc=al T
HEN GO TO 3064
3050 FOR t=w TO LEN r$: IF r$(t)
<>" " THEN LET w=t: GO TO 3037
3052 NEXT t: GO TO 3036
3064 LET p$=r$: GO SUB pt: RETUR
N
3078 IF r$(si)<>" " THEN GO TO 3
064
3080 LET r$=r$(pr TO )+" ": GO T
O 3078
3499 REM
3500 REM ** ROUTINE DI STAMPA **
3501 REM
3502 LPRINT p$: RETURN
4100 IF arg>mxpg OR arg<si THEN
GO TO 468
4102 LET pa=arg: LET pc=pa*nc-nc
-mo: GO TO 1054
4110 LET t$(pc+mo33 TO pc+nc+mo)
=e$: GO SUB mp: GO TO 1500
4200 LET cf=al: INPUT "COSA TROV
O? ";c$: GO TO 4206
```

ZX SPECTRUM

Seguito listato Lucy writer.

```
4205 LET cf=si: INPUT "VECCHIO "
;c$,"NUOVO ";b$
4206 GO SUB pg: IF LEN c$>20 THE
N PRINT #si;AT si,al;w$;AT si,al
;"STRINGA TROPPO LUNGA (max 20)"
: PAUSE 60: GO TO cmd
4208 FOR f=si TO LEN c$: POKE f+
65426,CODE c$(f): NEXT f
4210 LET jj=LEN c$: POKE 65467,j
j-si: LET r1=si: LET c1=-si
4212 LET z1=pc+r1*mo+c1+7: POKE
65452,INT (z1/256): POKE 65451,z
1-256*PEEK 65452: LET z2=mxca-jj
-z1+6: POKE 65456,INT (z2/256):
POKE 65455,z2-256*PEEK 65456
4214 LET z=z1+z2-USR 65447-6: IF
 z<=mxca-jj-si THEN GO TO 4220
4216 PRINT #si;AT si,al;w$;AT si
,al;"STRINGA NON TROVATA.": PAUS
E 60: GO TO 1054
4220 LET pa=si+INT (z/nc): LET p
c=pa*nc-nc-mo: LET k=z-nc*INT (z
/nc): LET r1=INT ((k-si)/mo)+si:
 LET c1=k-mo*(r1-si)-si
4222 GO SUB pg: GO SUB mp
4224 PRINT AT r1,c1; INVERSE si;
t$(z)
4226 IF CODE INKEY$=7 THEN GO TO
 260
4228 IF INKEY$="a" THEN GO TO 42
12
4230 IF NOT (INKEY$="c" AND cf)
THEN GO TO 4226
4232 LET pos=pc+r1*mo+c1+si: IF
LEN b$<jj THEN LET n=jj-LEN b$:
GO SUB 1142: GO TO 4236
4234 LET a$=w$( TO LEN b$-jj): G
O SUB 2034
4236 LET t$(pc+r1*mo+c1+si TO pc
+r1*mo+c1+LEN b$)=b$: GO TO 4222
5000 CLS : PRINT AT 10,13; FLASH
 si;"SAVING": GO SUB 5020: GO SU
B 5022
5002 IF r$="save" THEN SAVE n$CO
DE ind,qua: GO TO restart
5004 GO SUB 5028: SAVE *"m";j;n$
CODE ind,qua: GO TO restart
5006 CLS : PRINT AT 10,12; FLASH
 si;"LOADING": RANDOMIZE USR 655
04: GO SUB 5020: LET qua=mxca
5008 IF r$="load" OR r$="append"
 THEN LOAD n$CODE ind,qua: GO TO
 restart
5010 GO SUB 5028: LOAD *"m";j;n$
CODE ind,qua: GO TO restart
5012 CLS : PRINT AT 10,13; FLASH
 si;"APPEND": INPUT "Da quale pa
gina? ";k: IF k<si OR k>mxpg THE
N GO TO 5012
5014 GO SUB 5020: LET i=k*nc-nc+
si: LET ind=ind+i-si: LET qua=mx
ca-i: GO TO 5008
5020 INPUT "Nome del testo (max
10) ", LINE n$: LET ind=6+PEEK 2
3627+256*PEEK 23628: RETURN
5022 POKE 65451,6: POKE 65452,al
: POKE 65455,224: POKE 65456,91:
 POKE 65467,2: POKE 65427,36: PO
KE 65428,42: POKE 65429,36
5024 LET qua=mxca-USR 65447-si:
IF qua>mxca-mo31 THEN LET qua=mx
ca
5026 RETURN
5028 INPUT "Quale microdrive? ";
j: IF j<si OR j>8 THEN GO TO 502
8
5030 RETURN
5032 CLS : PRINT AT 10,13; FLASH
 si;"FORMAT": GO SUB 5028: STOP
: FORMAT "m";j;"testi": GO TO 50
36
5034 LET j=arg: IF j<si OR j>8 T
HEN GO TO 468
5036 CLS : CAT j: PAUSE al: GO T
O restart
5038 CLS : PRINT AT 11,12; FLASH
 si;"ERASING": GO SUB 5020: GO S
UB 5028: INPUT INVERSE si;" CONF
ERMI? "; INVERSE al;" "; LINE a$
: IF a$<>"si" THEN GO TO restart
5040 ERASE "m";j;n$: GO TO resta
rt
5050 INPUT "Ink? ";j;" Paper? ";
k
5052 BORDER k: INK j: PAPER k: P
OKE 23624,j+8*k: GO SUB mp: GO T
O cmd
7070 CLS : PRINT INVERSE 1;"INFO
RMAZIONI"
7072 PRINT '"NOME DEL TESTO    :
";n$
7076 PRINT '"PAGINA CORRENTE   :
";pa
7078 PRINT '"COORDINATE CURSORE

:  ";r;" - ";c
7080 PRINT '"PARAMETRO DI TABULA
ZIONE :   ";tab
7082 PRINT '"CARATTERI PER LINEA
 :   ";cpl
7084 PRINT #si;"PREMI UN TASTO P
ER RITORNARE"
7094 PAUSE 0
7095 GO TO 1054
8000 DIM t$(23520): LET mxca=LEN
 t$: LET qs=8518: LET pt=3500: L
ET pg=8506: LET mp=1160: LET cmd
=402: LET al=0: LET si=1: LET pr
=2: LET mo31=31: LET mo=32: LET
mo33=33: LET nc=672
8010 DIM w$(mo): LET restart=950
: LET mxpg=mxca/nc: LET pa=si: L
ET pc=-mo: LET tab=5: LET cpl=mo
: DIM e$(cpl+si)
8100 FOR f=al TO 7: READ a$: POK
E USR "♪"+f,VAL a$: NEXT f
8102 FOR f=al TO 7: READ a$: POK
E USR "■"+f,VAL a$: NEXT f
8104 POKE 23609,40
8200 DATA "0","12","10","8","56"
,"120","48","0","0","0","255","2
55","255","255","0","0"
8202 DATA "42","75","92","1","6"
,"0","9","1","224","91","17","14
7","255","26","237","177","192",
"229","197","6","1","120","254",
"0","40","9","43","19","35","26"
,"190","32","5","16","248","193"
,"225","201","193","225","24","2
24"
8204 DATA "42","75","92","1","6"
,"0","9","1","224","91","62","14
4","237","177","201","42","75","
92","1","6","0","9","1","224","9
1","62","32","119","35","11","12
0","177","200","24","246","33","
0","0","17","0","0","1","0","0",
"237","184","201"
8400 RETURN
8506 PRINT #si;AT al,al;"■■■■■■■■
■■■■■■■■■■■■■■■■■■■■■■■■■■";w$;AT
 si,al;"PAGE: ";pa
8510 IF pa=mxpg THEN PRINT #si;A
T si,10; INVERSE si;" ** FINE DE
L TESTO ** "
8512 IF pa=si THEN PRINT AT al,a
l; INVERSE si;" ** INIZIO DEL TE
STO ** "; INVERSE al;"         "
8514 RETURN
8520 CLS
8522 INPUT "Dalla pag.? ";aa;" A
lla pag.? ";bb
8524 IF aa<si OR aa>mxpg OR bb<s
i OR bb>mxpg OR bb<aa THEN GO TO
 8522
8528 LET aa=aa*nc-nc+si: LET bb=
bb*nc: CLS : RETURN
8530 POKE 65526,INT (v1/256): PO
KE 65525,v1-256*PEEK 65526
8532 POKE 65529,INT (v2/256): PO
KE 65528,v2-256*PEEK 65529
8534 POKE 65532,INT (v3/256): PO
KE 65531,v3-256*PEEK 65532
8536 RETURN
```

SHARP MZ-700

Free sky videogame

Per gli appassionati di sci e possessori dello Sharp MZ-700 ora esiste Free sky

di Alberto Zampariolo

Questo programma è una simulazione del famoso sport, fatta ovviamente al computer. Il videogame si articola in tre specialità:

- discesa libera;
- slalom gigante;
- crazy slalom.

Esistono all'interno delle prime due specialità degli ostacoli da evitare (bandierine) e dei bonus che contribuiscono ad aumentare il punteggio.

Il giocatore ha a disposizione dieci possibilità di errore (cadute) in tutte le manche, dopo di che ha termine il gioco (il punteggio relativo alla manche in corso, quando vengono effettuate le dieci cadute, non viene conteggiato ai fini del punteggio totale del gioco).

Listato 1 - Il programma Free sky.

```
10 REM////////////////\\\\\\\\\\\\\\\\
20 REM//////                    \\\\\
30 REM/////    (c) By Z. Soft.    \\\\
40 REM////                         \\\
50 REM\\\\    Alberto Zampariolo    ///
60 REM\\\\\                       ////
70 REM\\\\\         MONZA         ////
80 REM\\\\\\\                   //////
90 REM\\\\\\\\\\\\\\\\////////////////
100 POKE 89,240
110 '++++++++++++++++++++++++++++
120 '++++++++++++++++++++++++++++
130 '++  INTESTAZIONE PUNTI ++++++
140 '++++++++++++++++++++++++++++
150 DIM FM$(10),FM(10)
160 FM(1)=10000
170 FM$(1)="ALBER"
180 FM(2)=7000
190 FM$(2)="Z.SOFT"
200 FM(3)=5000
210 FM$(3)="DELETOR"
220 PQ=3
230 '++++++++++++++++++++++++++++
240 '+++ FINE INTESTAZIONE PUNTI+
250 '++++++++++++++++++++++++++++
260 '****************************
270 '***  PRESENTAZIONE GENERALE*
280 '****************************
290 TEMPO 7
300 COLOR,,7,0:CLS:CONSOLE3,21,5,30
310 COLOR,,6,1:CLS:CONSOLE
320 FORA=$D800TO$D800+119:POKEA,240:NEXT
330 POKE $D000+49,$1A,$85,$13,$20,$6,$14
,$17,$1,$12,$5,0,$85,$10,$12,$20,$4,$15,
$3,$14,$9,$20,$E
340 CURSOR36,4:PRINT[7,0]"(c)▯▯▯▯B▯▯y▯▯▯
Z.▯▯▯▯S▯▯o▯▯f▯▯t▯▯w▯▯a▯▯r▯▯e"
350 CURSOR1,4:PRINT[7,0]"(c)▯▯▯▯B▯▯y▯▯▯Z
.▯▯▯▯S▯▯o▯▯f▯▯t▯▯w▯▯a▯▯r▯▯e"
360 FOR QS=0 TO 190
370 X=INT(SIN(QS)*20)+20
380 C=INT(RND(5)*7)+1
390 IF C=1 THEN 380
400 Y=INT(COS(QS)*24)+24
410 SET Y+15,X+7,C
420 POKE $A3A,ABS(X-Y)
430 USR(68)
440 NEXT QS:K1=1:J=4
450 FOR E3=1 TO 7
460 J=J+1:K1=K1+4
470 CURSORK1,J:PRINT[0,7] "F▯R▯E▯E▯▯S▯K▯
Y▯"
480 POKE$A3A,E3+4:USR(62):MUSIC"R2":NEXT
:FORO=1 TO 900:NEXT
490 FOR GL=1 TO 7
500 CONSOLE(13-(GL+4)),((GL+4)*2),(19-(G
L+7)),((GL+7)*2)
510 COLOR,,ABS(8-GL):CLS
520 FOR KQ=GL*7 TO 80
530 POKE $A3A,KQ:USR(68)
540 NEXT KQ
550 NEXT GL:COLOR ,,7,1:CONSOLE:USR(71)
```

Nel crazy slalom, invece, non esiste possibilità di caduta, ma si presenta il fattore ghiaccio: è una prova d'abilità che qualifica il giocatore per i successivi schermi di crescente difficoltà.

Ovviamente la difficoltà aumenta all'aumentare dei quadri.

Descrizione del programma

Dopo aver brevemente illustrato il funzionamento del programma sotto il punto di vista del gioco, si può passare ora alla spiegazione più globale del videogame Free sky come puro e semplice flow chart. Questo si divide in varie fasi, le quali si dividono a loro volta in routine con compiti specifici.
Ecco le divisioni.

- Fase anteriore alla giocata:
  - presentazione;
  - istruzioni del gioco;
  - inizializzazione delle variabili.
- Fase attiva di gioco:
  - ciclo primo;
  - ciclo secondo;
  - ciclo di For;
  - ciclo terzo quadro.
- Fase interattiva al gioco:
  - routine dei suoni delle label;
  - routine dei suoni degli ostacoli;
  - routine di visualizzazione punteggio.
- Fase posteriore alla giocata:
  - routine memorizzazione punti;
  - routine coppa;
  - ciclo di End.

Prima della spiegazione delle fasi di gioco, bisogna dire che nel listato, dato che non vengono stampati particolari caratteri, appaiono dei codici (esadecimali) in alcune istruzioni di stampa.
Esempio: (810 PRINT →→→→ →→→→→ FEC8C8C8 eccetera).
Questi codici vanno cercati nella tabella dei codici schermo nel manuale allegato a ogni Sharp. Importante in questo gioco, l'uso di routine in linguaggio macchina, che velocizzano molto l'esecuzione.

Per esempio: la routine del suono che permette di manipolare dati musicali molto più velocemente che con l'istruzione Music del Basic.

Spiegazione del listato

- Presentazione:
  - va dalla riga 290 fino alla 550;

Seguito listato Free sky.

```
560 '******************************
570 '**** FINE PRESENTAZIONE  *****
580 '******************************
590 'ooooooooooooooooooooooooooooo

600 'ooo  SCELTA ISTRUZIONI ooooooo
610 'ooooooooooooooooooooooooooooo
620 CLS:PRINT:PRINT:PRINT"■■■■■          VU
OI LE ISTRUZIONI (s/n)?"
630 GETVQ$:IF VQ$="" THEN 630
640 IFVQ$="N"THEN 730
650 IFVQ$="S"THEN GOTO 2790
660 IFVQ$<>""THEN 630
670 'ooooooooooooooooooooooooooo
680 'ooo FINE SCELTA ISTRUZIONI oo
690 'ooooooooooooooooooooooooooooo
700 'qqqqqqqqqqqqqqqqqqqqqqqqqqqqq
710 'qqqq INIZIO CICLO PRIMO qqqqq
720 'qqqqqqqqqqqqqqqqqqqqqqqqqqqqq
730 IF CN>5 THEN CN=0
740 CN=CN+1
750 IF CN<3THEN COLOR,,7,0:YA=240
760 IF CN>3THEN COLOR,,0,7:YA=143
770 CLS
780 PRINT
790 PRINT[5,0]"■■■■■■C2C8C8C8C8E9■■■■■■
F5C8C8C8E9"
800 PRINT[5,0]"■■■■■■C8C8C8C8C8E9■■■■■
■C8C8C8C8C8"
810 PRINT[5,0]"■■■■■■FEC8C8C8C8C9■■■■■
■FEC8C8C8C9"
820 PRINT:PRINT
830 PRINT"                  F5C8C8C8C8C8C8C
8C8C8C8C8E9
840 PRINT"                F5C8C8C8C8C8C8C8
C8C8C8C8C8C8E9
850 PRINT"        F5C8C8C8C8C8C8C8C8C8C8
C8C8C8C8C8C8C8C8C8C8C8C8E9
860 PRINT"    F5C8C8C8C8C8C8C8C8C8C8C8C
8C8C8C8C8C8C8C8C8C8C8C8C8C8E9
870 PRINT"C8C8C8C8C8C8C8C8C3C8C8C8C8C8
C8C8C8C8C8C8C8C8C8C8C8C8C8C8C8C8C8C8C8
C8C8C8C8
880 PRINT"C8C8C8C8C87070707070707070707070
707070707070707070C8C8C8C6C87070707070C8
C8C8C6"
890 PRINT"C8C8C8C87070700 6363636363636
363 63636363 7070700C8C8C8C870636370707070
C8C8"
900 PRINT"C8C8C8C87070700 63636363635363
 636363 707070C8C8C8C870706363637070700C8
C8C8"
910 PRINT"C8C8C870707070 63636363 63636363
 63 707070C8C8C8C8C870706363636363707070C8
C8C8"
920 PRINT"C8C8C8C87070707070707070707070
7070707070700C8C8C8C8C8C8707070707070707070
C8C8C8C8"
930 PRINT"C8C8C8C8C8C8C8C8C8C8C8C8C8C8C8
C8C8C8C8C8C8C8C8C8C8C8C8C8C8C8C8C8C8C8C8
C8C8C8C8
940 PRINT"C8C8C8C8C8 68686868   C8C8C8C8
```

- musicata.
- Istruzioni:
- sono quattro schermi; uno per ogni quadro più una sotto introduzione;
- primi tre schermi da 2770 a 3310;
- l'ultimo schermo da 4560 o 4750;
- musicate.
- Inizializzazione variabili:
- questo punto non ha un inizio precisto e definito in quanto ogni routine ha bisogno, prima di essere eseguita, una:
- inizializzazione delle variabili;
- azzeramento variabili da 3660 a 3700;
- inizializzo punti da 120 a 240.
- Routine dei suoni label e ostacoli:
- queste sono routine di suoni che intervengono quando lo sciatore prende il bonus o sbatte contro una bandierina;
- suoni label inizia a 2040 e finisce a 2120;
- suoni ostacolo inizia a 1920 e finisce a 2110.
- Routine visualizzazione punti:
- ogni volta che una specialità è terminata, questa routine dà la situazione del punteggio dei giocatori;
- inizia a 2530 e finisce a 2740.
- Memorizzazione punteggio:
- memorizza il punteggio e lo confronta con quelli ottenuti precedentemente facendo una classifica;
- inizia a 3360 e finisce a 3630.
- Routine coppa e ospedale:
- sono due routine che, rispettivamente, assegnano una coppa a chi riesce a superare il record e ricoverano in ospedale un giocatore che ha abbattuto troppi ostacoli;
- ospedale inizio 3730 fino 4190;
- coppa inizio 4220 fino 4520.
- Ciclo End:
- coordina le fasi posteriori alla fase attiva di gioco e da questo punto si decide se giocare ancora o terminare la partita;
- inizia 1690 finisce a 1890.

Seguito listato Free sky.

```
686868б8 C8C8C8 68686868 C8C8C8C8C8C8C8
"
950 PRINT"C8C8C8C8C8C8C8C8C8C8C8C8C8C8C8
C8C8C8C8C8C8C8C8C8C8C8C8C8C8C8C8C8C8
C8C8C8C8
960 PRINT"C8C8E9                 FEC8C8C8
C8C8C8C8C8C8C8C8C8C8C8C8C8C8C8C8C8"
970 PRINT"EDEDEDED                 EDEDED
EDEDEDEDEDEDEDEDEDEDEDEDEDEDEDEDED"

980 PRINT"EDEDEDEDED                  EDED
EDEDEDEDEDEDEDEDEDEDEDEDEDEDEDEDED"
990 PRINT"359595959595                 95
95959595959595959595J5959595J53535"
1000 PRINT"9595959595                   9
59595959595959595959595959595959595"
1000 PRINT"9595959595                   9
53595959595959595959595959595959595"
1010 NG=$D000+853:POKE NG,88
1020 CURSOR3,23:PRINT[7,4]"        S T A
 R T      "
1030 FOR C1=210 TO 60 STEP-3
1040 POKE $A3A,C1 :USR(68)
1050 NEXT C1
1060 FOR C1= 80 TO 200 STEP-3
1070 POKE $A3A,C1 :USR(68)
1080 NEXT C1
1090 GET B$:IF B$<>"" THEN 1110
1100 GOTO 1030
1110 CURSOR3,23:PRINT"
 "
1120 '@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
1130 '@@@@  INIZIO CICLO SECONDO @@
1140 '@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
1150 TEMPO 7
1160 TI$="000000":T$=TI$
1170 G=15:B=3:N=18:W=2:C=0:FQ=0:LA=0
1180 IF ((CN=2)+(CN=5))  THEN W=3
1190 IF ((CN=3)+(CN=6)) THEN GOTO 2170
1200 FOR V=B TO N STEP W:V1=40-(G+V)
1210 'UUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUU
1220 'UUUUU INIZIO CICLO DEL `FOR `UUUU
1230 'UUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUU
1240 IF LT>70  THEN LT=0
1250 IF LT<0 THEN LT=70
1260 USR(68):POKE$A3A,LT
1270 FOR F=1 TO V:PRINT"ED";:NEXT F
1280 FOR D=1 TO G:PRINT" ";:NEXT D
1290 FOR F=1 TO V1:PRINT"ED";:NEXT F
1300 GET U$:IF U$<>""THENW$=U$
1310 IF W$="⊞" THEN NG=NG+1: LT=LT+1
1320 IF W$="⊟" THEN NG=NG-1:LT=LT-1
1330 IF W$="◫" THEN NG=NG-40:LT=LT+10
1340 IF W$="◫" THEN NG=NG+80:LT=LT-10
1350 IF ((CN=2)+(CN=5)) THEN 1410
1360 M=(RND(5)*40)+$D000+960
1370 M1=(RND(5)*40)+$D000+960
1380 IF PEEK(M-40)=0 THEN POKE M+$800-40
```

Aut. Min. Conc.



ABBÒNATI, RISPARMIA, UNA PEUGEOT 205 E 1000

14 riviste di elettronica, informatica, personal computer, 14 modi per vincere subito! Abbònati compilando e inviando l'apposito coupon: parteciperai al grande Concorso Jackson '86 e potrai vincere ad estrazione una magnifica Peugeot 205 XR o uno splendido orologio Commodore Time (e in palio ce ne sono ben mille!) E in più l'abbonamento ti dà diritto, fino al 28.2.86, anche al 20% di sconto su tutti i libri della Biblioteca Jackson (di cui puoi richiedere il catalogo con il coupon stesso).

JACKSON FA LEGGE LEGGI JACKSON



Seguito listato Free sky.

```
,YA :POKE M-40,$3C:C=C+1:IFC>60 THEN 156
0
1390 IF PEEK(M1-40)=0 THEN POKE M1+$800-
40,YA :POKE M1-40,$AA
1400 GOTO 1460
1410 LA=LA+1
1420 IF LA=1 THEN POKE $D800+920+14,YA ,
YA,YA :POKE$D000+920+14,$AA,0,$3C:FQ=FQ+
1
1430 IF LA=12THEN POKE $D800+920+12,YA ,
YA,YA :POKE$D000+920+12,$3C,0,$AA:FQ=FQ+
1
1440 IF LA=24THEN LA=0
1450 IF FQ=16 THEN GOTO 1560
1460 IF NG>$D000+920 THEN NG=NG-40
1470 IF NG<$D000 THEN NG=NG+40
1480 L=PEEK(NG):IF ((L<>0)*(L<>88)*(L<>$
AA))THEN GOTO  1940

1490 L=PEEK(NG):IF L=$AA THEN BA=BA+1:GO
SUB 2060
1500 POKE NG,88
1510 '←←←←←← STAMPA CARATTERE →→→→→→→
1520 NEXT V:GB=B:B=N:N=GB:W=W*-1:GOTO 12
00:
1530 'UUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUU
1540 'UUUU FINE CICLO DEL FOR UUUUUUUU
1550 'UUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUU
1560 T$=TI$:COLOR,,7,0:USR(71):GOSUB 255
0
1570 MUSIC "R6"
1580 CURSOR6,23:PRINT[7,1]"           PUSH
  A  KEY          "
1590 GETNB$:IF NB$="" THEN 1590
1600 '@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
1610 '@@@@ FINE CICLO SECONDO @@@@@@
1620 '@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
1630 GOTO 730
1640 'qqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqq
1650 'qqqq  FINE CICLO PRIMO   qqqqqqq
1660 'qqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqq
1670 COLOR,,7,0
1680 '[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[
1690 '[[[[[  CICLO DELLA FINE   [[[[[
1700 '[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[
1710 FOR AL=1 TO 300
1720 SX=INT(RND(4)*79)+1
1730 N8=INT(RND(4)*49)+1
1740 N9=INT(RND(4)*7):SET SX,N8,N9
1750 POKE$A3A,N8+N9:USR(68):NEXT:USR(62)
1760 GOSUB 3760
1770 CONSOLE0,10,0,39
1780 GOSUB 3390
1790 CONSOLE
1800 IF FM(1)=PU THEN 4240
1810 CLS:PRINT"@@@@@@@  VUOI FARE UN ALT
RA PARTITA (s/n)                  "
1820 GET BX$:IFBX$="" THEN 1820
1830 IFBX$="S" THEN GOSUB3680:GOTO 620
1840 IFBX$="N" THEN 1860
1850 IFBX$<>""  THEN 1820
1860 AQ$="D5R0D4D3E3D5R3B5R0B4B3+C3B5R3B
5+D4+D3+C3B6A5B4B3A3G5R3D5R0D4D3E3D5R3B5.
R0B4B3+C3B5R3B5+D4+D3+C3B6A5B4B3A3G6"
1870 TEMPO 7:PRINT"@@@@@          Z.Softwar
e ":MUSICAQ$:CLS:END
1880 '[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[
1890 '[[[[ FINE CICLO DELLA FINE  [[[[[
1900 '[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[
1910 '::::::::::::::::::::::::::::::::
1920 '::::  SUONO INCIDENTE    ::::::
1930 '::::::::::::::::::::::::::::::::
1940 FOR X1=1 TO 10
1950 FOR Y1=0 TO 80 STEP 2
1960 POKE $A3A,Y1:USR(68)
1970 NEXT Y1:USR(71):NEXT X1
1980 OS=OS+1: IF OS>=10 THEN GOTO1670
1990 GOTO 1490
2000 '::::::::::::::::::::::::::::::::
2010 '::: FINE SUONO INCIDEDNTE :::::
2020 '::::::::::::::::::::::::::::::::
2030 '*****************************
2040 '****  SUONO DEL BONUS  ******
2050 '*****************************
2060 FOR X1=1 TO 5

2070 FOR Y1=80 TO 0 STEP-10
2080 POKE $A3A,Y1:USR(68)
2090 NEXT Y1:USR(71):NEXT X1
2100 RETURN
2110 '******************************
2120 '*** FINE SUONO BONUS  *********
2130 '******************************
2140 '..............................
2150 '...  CICLO TERZO QUADRO .......
2160 '.............................
2170 N=1:FT=34:CURSOR0,24
2180 FOR I3=1 TO 300
2190 G=SIN(I3/2)*9
2200 FT=FT+N:IF FT>65 THEN N=N*-1
```

SHARP MZ-700

Seguito listato Free sky.

```
2210 IF FT<34 THEN N=N*-1
2220 LS=INT(RND(1)*10)
2230 IF LS<>5  THEN 2260
2240 SP=INT(RND(9)*33)
2250 CURSORSP,21: PRINT"C8C8C8C8C8C8C8C8
▥▥"
2260 NQ=FT-24:A9=INT(RND(1)*5)+2
2270 SET G+NQ,48,A9+1 :SETG+FT,49,A9
2280 SETG+NQ+1,49,A9-1 :SETG+FT+1,49,A9
2290 SET G+NQ+2,49,A9:SETG+FT+2,49,A9+1
2300 SETG+NQ+3,49,A9:SETG+FT+3,48:PRINT
2310 GET XC$:IF XC$="" THEN XC$=CHR$(PEE
K(95))
2320 IF XC$="▤" THEN NG=NG+1:GOTO2360
2330 IF XC$="▥" THEN NG=NG-40:GOTO2360
2340 IF XC$="▤" THEN NG=NG-1:GOTO 2360
2350 IF XC$="▥" THEN NG=NG+40
2360 NG=NG+40
2370 POKE$A3A,ABS(G*7) :USR(68)
2380 IF NG>$D000+1000 THEN NG=NG-40
2390 IF NG<$D000  THEN NG=NG+40
2400 IF NG<$D000+1000 THEN NG=NG-40
2410 IF PEEK(NG)<>$43 THEN 2450
2420 IF INT(RND(6)+.5)=1 THEN NG=NG+INT(
RND(6)*6)+2:GOTO 2440
2430 NG=NG-INT(RND(6)*6)+2
2440 GOTO 2460
2450 AW=PEEK(NG):IF((AW<>0)*(AW<>88))THE
N OS=1P: GOTO 1940
2460 POKE NG,88
2470 NEXTI3
2480 GOTO730
2490 '.............................
2500 '.. FINE CICLO TERZO QUADRO ....
2510 '.............................
2520 '£££££££££££££££££££££££££££££££
2530 '£££ VISUALIZZAZIONE PUNTI £££££
2540 '£££££££££££££££££££££££££££££££
2550 COLOR,,1,7:PRINT"▤"
2560 SO=100*(INT((BA-OS)*100/VAL(T$)))
2570 PU=PU+SO
2580 IF PU<0 THEN 4810
2590 IF ((CN=1)+(CN=4)) THEN XZ$="
    DISCESA LIBERA "
2600 IF ((CN=2)+(CN=5)) THEN XZ$="
    SLALOM GIGANTE "
2610 IF ((CN=3)+(CN=6)) THEN XZ$="
    CRAZY SLALOM     "
2620 PRINT:PRINTXZ$
2630 PRINT
2640 PRINT[6,3]"▤▤▤▤▤▤▤SCIATORI ",10-OS
2650 CURSOR4,6:PRINT[6,3] " CADUTE TOTAL

I "
2660 CURSOR7,8:PRINT[6,3]" "OS" "
2670 CURSOR23,6:PRINT[6,1]" BANDIERINE
  "
2680 CURSOR26,8:PRINT[6,1]" "BA" "
2690 PRINT:PRINT:PRINT[6,2]"▤▤▤▤▤ TEMPO
RELATIVO  ";T$ " SECONDI "
2700 PRINT:PRINT[6,2]"▤▤▤▥▥▤▤        PUNTI
 FINALI    " SO" "
2710 PRINT:PRINT[7,4]"▤▤▤▤▤▤▤▤▥          PUN
TI TOTALI       ▥▤▤▤▤▤▤▤▤▤▤▤▤▤▤▤▤▤▤▤▤▤▤▤
▤▤▤▤▤▤////////       " PU "      \\\\\\\\
"
2720 COLOR,,7,1:RETURN
2730 '£££££££££££££££££££££££££££££££
2740 '££££  FINE VISUALZZAZIONE PUNTI££
2750 '£££££££££££££££££££££££££££££££££
2760 'xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
2770 'xxx ISTRUZIONI GIOCO xxxxxxxxxx
2780 'xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
2790 CLS:PRINT:PRINT:PRINT
2800 PRINT"  IL GIOCO SI COMPONE DI 3 SP
ECIALITA`"
2810 PRINT:PRINT" 1 ]  DISCESA LIBERA "
2820 PRINT:PRINT" 2 ]  SLALOM GIGANTE "
2830 PRINT:PRINT" 3 ]  CRAZY SLALOM   "
2840 PRINT:PRINT"              A T T E N Z I
 O N E "
2850 PRINT:PRINT"  ESISTONO  OSTACOLI
SULLA PISTA"
2860 PRINT:PRINT" ED  ESISTONO  DELLE LA
BEL          "
2870 PRINT:PRINT:PRINT[0,7]"▤▤▤▤▤▤▤▤▤▤▤▤
▤ PUSH A KEY "
2880 GOSUB 3240
2890 CLS: PRINT[0,7] "▥▥▥▤▤▤▤▤▤▤▤      FI
RST SCREEN   "
2900 PRINT"▥▥     Questa e` la discesa l
ibera "
2910 PRINT"▥▥▥   DUE TIPI DI F5E9  L A B
 E L  FEC9  KEY "
2920 PRINT"▥▥    1 ]  LABEL BONUS     "
2930 PRINT"▥    2 ]  LABEL DIDUS      "
2940 PRINT"▥▥▥▥      Movimenti  "
2950 PRINT[1,7]TAB(10)"▥▥▥     PUSH A KEY
   ":POKE$A3A,60:USR(68)
2960 POKE $D800+12*40+23,243,243,243:POK
```

Seguito listato Free sky.

```
E$D000+12*40+24,$AA
2970 POKE $D800+14*40+23,243,243,243:POK
E$D000+14*40+24,$3C
2980 POKE$D000+19*40+20,$C4,0,0,0,$C3
2990 POKE$D000+17*40+22,$C2
3000 POKE$D000+21*40+22,$C1
3010 SET 44,39:SET 44,38
3020 SET 45,39:SET 45,38
3030 GETC$: IF C$="" THEN 2960
3040 CLS:PRINT[0,7] "@@@@@@@@@@@@@@     SE
COND SCREEN  "
3050 PRINT"@@    SLALOM GIGANTE  ?@\#↓↑
"
3060 PRINT"@         DUE TIPI DI LABEL      "
3070 PRINT"@         MA DISPOSTE IN MODO CA
TEGORICO"
3080 PRINT"@         A SLALOM GIGANTE "
3090 PRINT"@@      I movimenti sono gli st
essi "

3100 PRINT[0,7] "@@@@@@@@@@@@@@@   I     P U
 N T I   "
3110 PRINT"@         SONO DATI DALLA COMINAZ
 IONE DI
3120 PRINT"@         DEL TEMPO / OSTACOLI /
 BONUS "
3130 GOSUB 3240
3140 CLS
3150 COLOR,,0,7
3160 CLS
3170 CONSOLE3,19,3,34
3180 COLOR,,7,0
3190 CLS
3200 GOSUB 4570
3210 CONSOLE
3220 GOSUB 3240
3230 GOTO730
3240 GET C$:IF C$<>"" THEN RETURN
3250 Q5=Q5+1
3260 POKE $A3A,Q5:USR(68)
3270 IF Q5=> 70 THEN Q5=INT(RND(19)*70)+
20
3280 CURSOR1,1:PRINT "63@";:USR(62)  :PR
INT" ";
3290 CURSOR1,23:PRINT "63@";:USR(62)  :P
RINT" ";:CURSOR38,23:PRINT "63@";:USR(62
)  :PRINT" ";
3300 CURSOR38,1:PRINT "63@";:USR(62)  :P
RINT" ";
3310 GOTO 3240
3320 'xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
3330 'xxxx FINE ISTRUZIONI xxxxxxxxxxx
3340 'xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
3350 '????????????????????????????????
3360 '??? MEMORIZZAZIONE PUNTI ???????
3370 '???  E GRADUATORIA       ???????
3380 '????????????????????????????????
3390 REM
3400 COLOR,,4,2:CLS:PQ=PQ+1:FM(PQ)=PU:GO
TO 3540
3410 CLS:PRINT"       C8C8E9  C8C8C8  F5C
8E9  F5C8E9  C8C8E9  C8C8E9
3420 PRINT"       C8 FEE9 C8      C8     C8
C8  C8 FEE9 C8 FEE9
3430 PRINT"       C8 F5C9 C8C8     C8     C8
 C8  C8 F5C9 C8  C8
3440 PRINT"       C3F5C9  C8      C8     C8
C8  C8 FEE9 C8  C8
3450 PRINT"       C6FEE9  C8C8C8  FEC8C9
 FEC8C9  C8  C8 C8C8C8C9
3460 PRINT"       C8 FEE9
3470 IF PQ=1 THEN GOTO 3530
3480 FOR N=1 TO PQ
3490 MAX=FM(N):FAP$=FM$(N)
3500 FOR VQ=N TO PQ
3510 IF MAX<FM(VQ) THEN MAX=FM(VQ):FAP$=
FM$(VQ):FM(VQ)=FM(N):FM$(VQ)=FM$(N):FM(N
)=MAX:FM$(N)=FAP$
3520 NEXTVQ,N:IF PQ>5 THEN PQ=4
3530 FOR JX=1 TO PQ:PRINT:PRINT"  ";:PRI
NTUSING"&                &";FM$(JX);:PRINT
USING"#############";FM(JX):PRINT:NEXT
JX:GOTO3590
3540 PRINT"@@@           NOME          PUNTE
GGIO":PRINT:PRINT
3550 INPUT FM$(PQ)

3560 PRINT"@@@@@@@@@@@@@@@@@@@@@@";FM(PQ):
MUSIC"R6"
3570 CONSOLE
3580 GOTO 3410
3590 PRINT[0,7]"@               PUSH A KEY
        "
3600 GET PL$: IF PL$="" THEN 3600
3610 COLOR,,7,0:RETURN
3620 '????????????????????????????????
3630 '????  FINE MEM.PUNTI ??????????
3640 '????????????????????????????????
3650 'C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0
C0C0C0C0C0C0C0C0C0C0C0C0C0C0
```

Seguito listato Free sky.

```
3660 'C0C0C0C0 AZZERA VARIABILI C0C0C0C0
C0C0C0C0C0
3670 'C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0
C0C0C0C0C0C0C0C0C0C0C0C0C0
3680 CN=0:LA=0:FQ=0:OS=0:PU=0:BA=0:T$="0
":SU=0
3690 'C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0
C0C0C0C0C0C0C0C0C0C0C0C0C0
3700 'C0C0C0C0 FINE AZZERA VARIABILI  C0
C0C0
3710 'C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0
C0C0C0C0C0C0C0C0C0C0C0C0C0
3720 '=================================
3730 '==== RICOVERO IN OSPEDALE  ====
3740 '=================================
3750 RETURN
3760 CLS
3770 COLOR,,5,1
3780 CLS
3790 PRINT" ▯    ti sei rotto una gamba
3800 PRINT" ▯    e le tue condizioni
3810 PRINT" ▯    sono pessime
3820 PRINT" ▯    I L  T U O  R I C O V E
R O
3830 PRINT" ▯         E' U R G E N T E
3840 COLOR,,7,1
3850 PRINT" ▯
3860 PRINT"          F5C8C8C8C8C8C8C8C8C8
C8C8C8C8C8C8C8C8E9
3870 PRINT"          C8C8H O S P I T A L C8
C8
3880 PRINT"          C8C8C8C8C8C8C8C8C8C8
C8C8C8C8C8C8C8C8C8
3890 PRINT"          C8C8++C8C8C8++C8C8++C8
C8C8++C8C8
3900 PRINT"          C8C8++C8C8C8++C8C8++C8
C8C8++C8C8
3910 PRINT"          C8C8C8C8C8C8C8C8C8C8
C8C8C8C8C8C8C8C8C8
3920 PRINT"          C8C8C8C8C8C8C8C8C8C8
C8C8C8C8C8C8C8C8C8
3930 PRINT"          C8C8  C8C8C8C8   FEC8C
8C8  C8C8C8
3940 PRINT"          C8C8  G8C8C8     C8C8
8  C8C8C8
3950 PRINT"          C8C8C8C8C8C8C8    C8C
8C8C8C8C8C8C8
3960 PRINT"          C8C8C8C8C8C8C8    C8C
8C8C8C8C8C8C8
3970 PRINT"                    F5C8C8C8C8C9
3980 PRINT"                    F5C8C8C8C8C9
3990 FOR M1=1 TO 18 STEP3
4000 FOR M=1 TO 49
4010 SET 3+M1,M:SET 78-M1,M
4020 POKE $A3A,140-M
4030 USR(68)
4040 NEXT M
4050 FOR M=1 TO 79
4060 POKE $A3A,90 +M
4070 USR(68)
4080 SET M,1+M1 :SET M,49-M1
4090 NEXT M
4100 NEXT M1
4110 USR(62)
4120 PRINT"▯▯▯▯▯▯▯▯▯▯▯

4130 PRINT"                 DAI IL TUO NOME

4140 PRINT"               OSSERVA IL PUNTEGGI
O
4150 PRINT[0,7]"               PUSH A KEY

4160 GET F$:IF F$="" THEN 4160
4170 RETURN
4180 '=============================
4190 '==== FINE RICOVERO IN OSPEDALE =
4200 '================================
4210 '$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
4220 '$$$$ ASSEGNAZIONE COPPA $$$$$$$$
4230 '$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
4240 CLS
4250 CONSOLE
4260 COLOR,,7,4
4270 CLS
4280 PRINT:PRINT
4290 PRINT"              HAI BATTUTO IL RECOR
D"
4300 PRINT
4310 PRINT"              SEI UN CAMPIONE ????
?? "
4320 PRINT
4330 PRINT"          LA Z.SOFTWARE  TI OFFR

E   "
4340 PRINT
4350 PRINT"          ******:   ZED CUP     :*
*****
4360 PRINT"           ******              **
```

Seguito listato Free sky.

```
*****
4370 PRINT"          ********  $$$$$ *****
***
4380 PRINT"        ******************
*
4390 PRINT"            *************
4400 PRINT"              ********
4410 PRINT"                ****
4420 PRINT"                ****
4430 PRINT"             //////\\\\\\
4440 PRINT"            //////  \\\\\\
4450 PRINT"            \\\\\\  //////
4460 PRINT"             \\\\\\//////
4470 PRINT:PRINT
4480 PRINT[0,7]"B              PUSH  A  KEY

4490 GET N$:IF N$="" THEN 4490
4500 RETURN
4510 '$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
4520 '$$$$ FINE COPPA $$$$$$$$$$$$$$$
4530 '$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
4540 '-------------------------------
4550 '--- ISTRUZIONI TERZO QUADRO ---
4560 '-------------------------------
4570 TEMPO 7
4580 CONSOLE
4590 CLS
4600 PRINT:PRINT:PRINT[3,1] "BBBBBF5C8E9
 F5C8E9 F5C8E9 F5C8 C8"
4610 MUSIC "A2+C1"
4620 PRINT[7,1]"BBBBBC8BBBC8BC9BC8BC8BFE
9BC8"
4630 MUSIC "D1+B1"
4640 PRINT[6,1]"BBBBBFEC8BBC89E9BC8D8C8B
F5E9BC8 :PRINT
4650 MUSIC "DADA"
4660 PRINT:PRINT:PRINT"BBBBBBBBBBBB FE  D
5C1D8D5C1 C1D8D5C1E9F5D5"
4670 PRINT"BBBBBBBBBBBBB E9 D5C1C2D5C1 C1
 D5C1  D5"
4680 PRINT"BBBBBBBBBBBBB C9 D5C1 D5C1 C1
D5C1  C5"
4690 PRINT"BBBBBBBBBBB  C9  D8    D8 D8
4700 PRINT:PRINT:PRINT[7,3]  "BBBBBBBB
  SLALOM  DELLA  MORTE      "
4710 PRINT:PRINT"BBBBBBB Una solo caduta
 significa lc"
4720 PRINT:PRINT"BBBBBBBBBBBBB f i n e  d
e i gioco "
4730 PRINT:PRINT[0,6]  "BBBBBBBBBB  OK
S T A R T  ↑↓↑↓  "
4740 RETURN
4750 '-------------------------------
4760 '--- FINE ISTRUZIONI TERZO QUADRO-
4770 '-------------------------------
4780 'PPPPPPPP   PPPPPPPPPPPPPPPPPPPPP
4790 'PPP NOTAZIONE PUNTEGGIO NEGATIVOP
4800 'PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
4810 CLS
4820 COLOR,,3,3
4830 CONSOLE15,10,1,38

4840 CLS
4850 CONSOLE:PRINT"B":COLOR,,0,7
4860 PRINT:PRINT
4870 PRINT"B          IL TUO PUNTEGGIO  "
4880 PRINT"B           E' NEGATIVO   "
4890 PRINT"B         PERCIO' PERDI UNO "
4900 PRINT"B           SCIATORE        "
4910 PRINT
4920 PRINT:FOR PA=1 TO 500:NEXT PA
4930 PRINT[7,0]"B            PUSH A KEY
          "
4940 GET QM$:IF QM$="" THEN 4940
4950 PU=0:SO=0
4960 OS=OS+1
4970 IF OS=>10 THEN COLOR,,7,0:CLS:GOTO
 670
4980 CLS:GOTO 2590
4990 'PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
5000 'PP FINE NOT.PUNTI NEGATIVO PPPPP
5010 'PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
```

Indirizzamento della grafica per pixel

Gestiamo le capacità grafiche del QL

di Massimo Modelli

Come dice il proverbio "non tutte le ciambelle riescono col buco" e anche il QL non fa eccezione a questa regola, mostrando nella grafica alcune imperfezioni. Gli unici nei sono rappresentati dal sistema di riempimento delle figure chiuse e dalla gestione delle coordinate. È di questo secondo aspetto che intendiamo occuparci in questo ariticolo. Per l'indirizzamento dei comandi grafici viene utilizzato un sistema di coordinate cartesiane standard indipendente dal modo grafico usato, a quattro o a otto colori, e dalla posizione del pixel. È inoltre possibile, per mezzo del comando Scale, spostare l'origine di questo sistema di riferimento relativamente alla finestra usata e modificarne anche la scala. Purtroppo però non è possibile definire indipendentemente la scala sui due assi, ma solo globalmente.

Segmenti di uguale lunghezza (in cm) sullo schermo avranno sempre anche la stessa lunghezza rispetto alle coordinate grafiche, sia che siano verticali che orizzontali.

Se da una parte ciò fa sì che i quadrati risultino sullo schermo effettivamente... quadrati, indipendentemente dal modo grafico usato e compensando la diversa densità dei pixel nelle due direzioni, d'altra parte non permette di indirizzare la grafica per pixel.

Questa carenza può risultare importante, a esempio, dovendo disegnare delle figure non troppo grandi, in quanto risulteranno lievemente diverse a seconda della zona dello schermo, o meglio a seconda del valore delle coordinate in cui verranno disegnate. Nel caso in cui l'ampiezza verticale grafica di una finestra sia stata fissata uguale a quella in pixel, tale deformazione, dovuta agli arrotondamenti eseguiti dal sistema nel far corrispondere i valori delle coordinate ai singoli punti indirizzabili, risulterà presente solo nel senso della larghezza, modificando le proporzioni delle diverse parti.

È possibile sopperire a questa mancanza per mezzo di una formula che fornisca le coordinate grafiche corrispondenti a ogni singolo punto indirizzabile dello schermo. Consideriamo per primo il caso dell'alta risoluzione: per ricavare questa formula si deve innanzitutto valutare il rapporto: tra i pixel orizzontali e quelli verticali: tra il numero dei pixel che compongono una linea della stessa lunghezza, misurata nel sistema di coordinate grafiche, nelle due direzioni orizzontale e verticale. Il valore da noi trovato per questo rapporto, determinato misurando la larghezza grafica di finestre di dimensioni note, è 1,35. Se le coordinate grafiche sono state settate, con il comando Scale, in modo che l'altezza della finestra (rispetto alla grafica) sia uguale all'ampiezza verticale in pixel e l'origine sia nell'angolo inferiore sinistro della finestra (con Scale WH, 0,0; dove WH è il numero di pixel sull'asse verticale della finestra, per intenderci il secondo parametro usato con i comandi Window e Open), allora un pixel di coordinate (rispetto sempre all'angolo inferiore sinistro) M,N sarà individuato dalle coordinate grafiche:

X = M/1,35

e:

Y = N

Nel caso più generale in cui le coordinate grafiche siano state inizializzate con il comando Scale H, OrgX,OrgY si dovranno modificare le semplici formule viste sopra:

X = (M/Cx) + OrgX

e:

Y = (N/Cy) + OrgY

dove:

Cx = 1,35 ★ WH/H

e:

Cy = WH/H

È importante ricordare che sia OrgX che OrgY sono dei numeri con segno.

Nel caso si lavori in bassa risoluzione, cioè con otto colori, per adattare le formule prima viste è sufficiente definire al posto di Cx, Cx8 = Cx/2. Poiché sia Cx (o Cx8) che Cy non dipendono dal punto da individuare: in un eventuale programma che dovesse tracciare delle linee tra due pixel, potrebbero venire definiti una sola volta. In questo programma sarebbe comodo definire due funzioni che operino la conversione da pixel a grafica, a esempio:

```
DEF FN Xg (Xp,Cx,OrgX)
LOCAL X
X=(Xp/Cx) + OrgX
RETURN X
END DEF Xg
```

e:

```
DEF FN Yg(Yp,Cy,OrgY)
LOCAL Y
Y=(Yp/Cy)+OrgY
RETURN Y
END DEF Yg
```

Un altro sistema per indirizzare per punti, particolarmente utile e veloce quando si debbano disegnare figure composte da rettangoli, consiste nell'utilizzo del comando Block che permette un indirizzamento misto, per pixel e grafico. Il difetto di questo comando è che se il rettangolo esce dai limiti della finestra viene generato un messaggio di errore e l'eventuale programma si ferma, cosa che non accade con i comandi di pura grafica. ■

Chess training

Come valutare e migliorare il proprio livello nel nobil gioco

di Sergio Borsani

Il programma Msx di questo mese è legato al gioco degli scacchi. La rivista **Personal Software** non è nuova alla pubblicazione di programmi su questo argomento, anzi ricordiamo di aver iniziato la collaborazione proprio con un Archivio scacchistico per il TI99/4A. Ma il programma attuale, vuoi per le diverse finalità, vuoi per il diverso computer al quale è dedicato, non ha proprio niente in comune con il suo lontano parente.

Chi conosce il Chess Challenger a dieci livelli sa che oltre a giocare a scacchi con una buona conoscenza delle aperture, è in grado di proporre 60 partite celebri durante le quali ci si può mettere nei panni di un grande maestro e confrontare le proprie mosse con quelle che egli ha realmente giocato. Questo Chess training sostanzialmente emula il Chess Challenger limitatamente a questa funzione operativa. Nel programma sono memorizzate 12 partite che possono essere giocate, a scelta, con il bianco o con il nero. Lo svolgimento viene seguito su una grande scacchiera che appare sullo schermo, dove le mosse si susseguono alla pressione di un tasto, mentre sulla destra appare una notazione consistente nelle coordinate della casa di partenza e in quelle della casa d'arrivo (a esempio: e2e4).

Le prime volte, quando non si conoscono ancora le partite, il programma fornisce indicazioni sul livello di gioco, sulle capacità di analisi e sul grado di conoscenza delle aperture; in seguito, quando verrà a mancare la novità, costituirà comunque un buon esercizio per la memoria e per aumentare il bagaglio d'esperienza.

La grafica

Come i lettori sanno, i sistemi Msx utilizzano un microprocessore per il video uguale o analogo a quello del TI99/4A. Tuttavia, il modo di gestire la Vdp Ram e le istruzioni grafiche del Basic, consentono nei primi possibilità notevolmente maggiori concedendo al TI Extended Basic solo una più sofisticata gestione degli sprite. Infatti con un computer Msx si ha a disposizione uno schermo grafico e le comuni istruzioni Line, Circle, eccetera o le meno immediate Vpeek e Vpoke; per di più si può mescolare testo e grafica semplicemente considerando lo schermo come un file aperto in modo output. L'alta risoluzione è limitata solo dal fatto che a ogni sequenza di otto pixel, corrispondenti a un byte di memoria, non possono essere associati più di due colori.

Dal momento che nel programma si deve rappresentare una scacchiera con i pezzi bianchi e quelli neri, si deve disporre di due colori di foreground e di due di background avendo cura, come s'è detto, che ogni byte non ne debba sopportare più di due alla volta. Ciò si ottiene imponendo alle caselle che formano la scacchiera una larghezza che abbia una misura, in pixel, multipla di otto. Considerando poi che un rapporto di 1:1 tra base e altezza, per la diversa distanza tra i pixel lungo le due direzioni orizzontale e verticale, dà luogo a un rettangolo invece che a un quadrato, si comprenderà come le dimensioni della scacchiera diventino quasi una scelta obbligata. Nel programma, infatti, ogni

Figura 1 - Immagine d'apertura del programma, Chess trainig contiene in memoria 12 partite celebri. Il concorrente , con il bianco o con il nero, si mette nei panni di un celebre campione cercando di indovinare le mosse che questi ha realmente eseguito.



Listato 1 - Il programma Chess training.

```
10 REM     c h e s s   t r a i n i n g
20 REM
30 REM           by Sergio Borsani
40 REM                 (1985)
50 SCREEN2,3 : COLOR 15,4,4 : DIM M$(100
)
60 DATA 00,00,00,00,00,00,00,00,00,00,01
,03,03,07,0F,0F,01,09,0B,0F,1F,3F,7F,FF,
9F,BF,FF,FF,FF,FF,FF,FF
70 DATA 00,80,80,E0,D8,F6,FB,FD,FE,FE,FF
,FF,FF,FF,FF,DF,00,00,00,00,00,00,00,80,
C0,C0,60,60,B0,B0,D8,D8
80 DATA 1F,1F,3F,3F,3F,3F,3D,1D,01,00,00
,00,00,00,00,00,FF,FF,FF,F8,E3,C3,C3,C7,
87,0F,1F,3F,3F,7F,7F,7F
90 DATA DF,BF,7F,7F,FF,FF,FF,FF,FF,FF,FF
,FF,FF,FF,FF,FF,E8,EC,EC,F4,F6,F6,F6,F6,
F7,F7,FB,FB,FB,FB,FB,FB
100 DATA 00,00,00,00,00,00,00,00,00,00,0
0,00,00,00,00,00,7F,7F,3F,3F,1F,00,1F,07
,03,03,07,1F,3F,7F,7F,7F
110 DATA FF,FF,FF,FF,FF,00,FF,FF,FF,FF,F
F,FF,FF,FF,FF,FF,FB,FB,FB,FB,FB,00,FC,F0
,E0,E0,F0,FC,FE,FF,FF,FF
120 FOR I=1 TO 6 : A$="" : FOR J=1 TO 32
 : READ X$ : A$=A$+CHR$(VAL("&H"+X$)): N
EXT J : SPRITE$(I)=A$ : NEXT I
130 FOR Y=0 TO 160 STEP 40 : FOR X=0 TO
96 STEP 32 : IF (X/32+Y/40)/2=INT((X/32+
Y/40)/2) THEN C%=4 ELSE C%=15
140 LINE(X,Y)-(X+31,Y+39),C%,BF : NEXT X
 : NEXT Y
150 I=0 : FOR Y=50 TO 114 STEP 32 : FOR
X=30 TO 62 STEP 32 : I=I+1 : PUT SPRITE
I,(X,Y),10 : NEXT X : NEXT Y
160 OPEN "GRP:" FOR OUTPUT AS #1
170 PRESET(136,32) : PRINT #1,"CHESS TRA
INING"
180 PRESET(184,72) : PRINT #1,"by" : PRE
SET(136,96) : PRINT #1,"Sergio Borsani"
190 PRESET(136,168) : PRINT #1,"premi un
 tasto" : PRESET(136,176) : PRINT #1,"pe
r cominciare"
200 K$=INKEY$ : IF K$="" THEN 200 ELSE C
LS
210 FOR J=1 TO 6 : PUT SPRITE J,(-32,209
) : NEXT J : SCREEN 2,2
220 DATA 0,0,0,0,0,0,255,255,255,255,255
,255,0,0,0,0,0,128,192,224,240,248,252,2
54,255,254,252,248,240,224,192,128
230 DATA 60,126,255,255,255,255,126,60
240 DATA 0,3,15,31,63,60,120,120,120,120
,0,0,0,0,1,1,0,192,240,248,252,60,30,30,
30,30,60,60,120,240,224,224
250 DATA 3,3,3,3,3,3,3,0,0,1,3,3,1,0,0,0
,192,192,192,192,192,192,192,0,0,128,192
,192,128,0,0,0
260 RESTORE 220 : A$="" : FOR J=1 TO 32
 : READ N% : A$=A$+CHR$(N%) : NEXT J : SP
RITE$(1)=A$
270 A$="" : FOR J=1 TO 8 : READ N% : A$=
A$+CHR$(N%) : NEXT J : FOR J=1 TO 24 : A
$=A$+CHR$(0) : NEXT J : SPRITE$(2)=A$
280 A$=CHR$(254) : FOR J=1 TO 31 : A$=A$
+CHR$(0) : NEXT J : SPRITE$(3)=A$
290 A$="" : FOR J=1 TO 32 : READ N% : A$
=A$+CHR$(N%) : NEXT J : SPRITE$(4)=A$
300 A$="" : FOR J=1 TO 32 : READ N% : A$
=A$+CHR$(N%) : NEXT J : SPRITE$(5)=A$
310 PRESET(16,0) : PRINT #1,"SCEGLI UNA
DI QUESTE PARTITE" : Y%=16 : FG%=1 : GOT
O 390
320 PUT SPRITE 1,(8,Y%),6
330 K$=INKEY$ : IF K$="" THEN 330
340 IF K$=CHR$(13) THEN 370
350 IF K$=CHR$(30) AND Y%>12 THEN Y%=Y%-
8 : GOTO 320
360 IF K$=CHR$(31) AND Y%<96 THEN Y%=Y%+
8 : GOTO 320 ELSE 330
370 PUT SPRITE 1,(-32,209) : CLS : P%=(Y
%-2)/8
380 ON P% GOTO 390,400,410,420,430,440,4
50,460,470,480,490,500
390 RESTORE 1400 : IF FG%=1 THEN GOSUB 1
070 ELSE 520
400 RESTORE 1420 : IF FG%=1 THEN GOSUB 1
070 ELSE 520
410 RESTORE 1450 : IF FG%=1 THEN GOSUB 1
070 ELSE 520
420 RESTORE 1490 : IF FG%=1 THEN GOSUB 1
070 ELSE 520
430 RESTORE 1510 : IF FG%=1 THEN GOSUB 1
070 ELSE 520
440 RESTORE 1540 : IF FG%=1 THEN GOSUB 1
070 ELSE 520
450 RESTORE 1570 : IF FG%=1 THEN GOSUB 1
070 ELSE 520
460 RESTORE 1590 : IF FG%=1 THEN GOSUB 1
070 ELSE 520
470 RESTORE 1610 : IF FG%=1 THEN GOSUB 1
070 ELSE 520
480 RESTORE 1640 : IF FG%=1 THEN GOSUB 1
070 ELSE 520
490 RESTORE 1660 : IF FG%=1 THEN GOSUB 1
070 ELSE 520
500 RESTORE 1690 : IF FG%=1 THEN GOSUB 1
070 ELSE 520
510 FG%=0 : Y%=10 : GOTO 320
520 CLS : READ B$,N$ : CT%=1
530 READ M$(CT%) : IF NOT M$(CT%)="MATTO
" AND NOT M$(CT%)="PATTA" AND NOT M$(CT%
)="ABB." THEN CT%=CT%+1 : GOTO 530
540 PRESET(8,40) : PRINT #1,"Premi:" : P
```

casella è larga 16 pixel e alta 20; per i colori dello sfondo si sono scelti l'azzurro chiaro e il blu, mentre i pezzi sono naturalmente bianchi e neri.

Durante il gioco i pezzi vengono continuamente cancellati da una casa e collocati in un'altra e si potrebbe pensare che, conoscendo l'organizzazione della Vdp Ram, non ci sia niente di meglio che usare la Vpoke per modificare la Bit Image dello schermo e la corrispondente tabella dei colori. Non è così. Per questo scopo risulta molto più semplice e veloce l'istruzione Draw che infatti, dopo un breve provino, è diventata la protagonista della parte grafica.

Il successo è dipeso in gran parte dalla possibilità di definire alcuni paramentri, come il codice del colore e le coordinate del punto di partenza del disegno, con delle variabili numeriche piuttosto che con delle costanti. In questo modo, inserita in una subroutine, l'istruzione Draw serve indifferentemente per disegnare un pezzo bianco o nero in qualsiasi casa della scacchiera. Allo scopo le variabili vengono inserite nella forma Draw "C=C%;" e Draw "BM=X%-;,=Y%" dove C% è il codice del colore e X% e Y% le coordinate dell'angolo in alto a sinistra della casa in cui va collocato il pezzo, espresse naturalmente in pixel.

Il testo

Come si è già detto, per scrivere un testo nel modo grafico è necessario aprire un file. L'istruzione assume la forma Open "GRP:" For Output AS =1, dove 1 indica il numero del file; in seguito, dopo aver stabilito il punto da cui deve iniziare la scritta con Preset(x,y), si potrà collocare una stringa sullo schermo grafico come se si scrivesse un record su un dispositivo di memoria esterno con Print =1,X$. Bisogna tener presente che, a differenza di quanto avviene nel modo testo, se si scrive una stringa sopra dei caratteri preesistenti, essa non sostituisce questi ultimi, ma si sovrappone a essi rendendo illegibile la frase; pertanto è necessario pulire preventivamente la parte dello schermo dove si desidera scrivere. Ancora una volta ciò potrebbe esser fatto con delle Vpoke, ma risulta senz'altro più agevole usare l'istruzione Line con l'opzione bf per disegnare un rettangolo riempito con il colore dello sfondo.

Queste considerazioni non esauriscono il discorso sulla parte del programma riguardante il testo. Infatti in essa rientra non solo la scrittura, ma anche la funzione di data entry. In questo caso il problema è che con lo schermo grafico non si può usare l'istruzione Input che pertanto deve essere sostituita da una routine che svolga una funzione analoga. La routine presente nel programma rappresenta addirittura un miglioramento rispetto all'istruzione Input, in quanto consente di definire la lunghezza della stringa e anche il tipo dei caratteri che si devono accettare. Questa prerogativa risulta qui particolarmente utile, poiché, trattandosi di coordinate nella notazione scacchistica, si dovranno scrivere soltanto le lettere dalla a e i numeri dall'uno all'otto. La routine occupa le linee di programma dalla 820 alla 900 ed è incentrata sull'istruzione Inkey$, la quale scandisce la tastiera per scoprire quali tasti vengono premuti permettendo così la costruzione di una stringa carattere per carattere. Il cursore è uno sprite avente la forma di una lineetta posta al di sotto delle lettere; sono attivi i tasti di controllo del cursore ma, per semplicità, non lo sono né il tasto Bs per il Back space né i tasti Ins e Del dal momento che una stringa di solo quattro caratteri non necessita di sofisticate funzioni di editing.

La notazione delle mosse

Per personalizzare il programma e modificarlo sostituendo alle attuali partite, altre magari più attuali, è necessario conoscere il modo in cui vengono memorizzate le mosse. Prendiamo a esempio la prima partita: Anderssen-Kieseritzki, nota col nome de "L'immortale"; la ritroviamo nelle istruzioni Data a partire dalla linea 1400. È regola generale che i primi due dati si riferiscano ai due giocatori; seguono le mosse secondo la notazione che prevede quattro caratteri, due per le coordinate della casa di partenza e due per quella di arrivo. Fanno eccezione alcune mosse particolari come l'arrocco, la promozione e la presa di un pedone al passaggio che vanno considerate doppie e per le quali sono necessarie stringhe di otto caratteri. Quando il programma incontra una mossa di otto caratteri, interpreta i primi quattro, come una mossa normale e i secondi quattro, come una mossa aggiuntiva. Nell'arrocco i primi quattro caratteri indicano il movimento del Re, mentre i secondi quattro della torre; così "e1g1h1f1" rappresenta l'arrocco corto del bianco. Nel caso di promozione la mossa aggiuntiva assume una forma particolare: il primo carattere deve esse-

Seguito listato Chess training.

```
RESET(16,56) : PRINT #1,"[v] per vedere
la partita"
550 PRESET(16,72) : PRINT #1,"[b] per gi
ocare col bianco" : PRESET(16,88) : PRIN
T #1,"[n] per giocare col nero"
560 K$=INKEY$ : IF K$="" THEN 560
570 IF K$="v" OR K$="V" THEN S=0 : GOTO
600
580 IF K$="b" OR K$="B" THEN S=1 : GOTO
600
590 IF K$="n" OR K$="N" THEN S=-1 ELSE 5
60
600 REM TABELLA
610 DATA 2,3,4,5,6,4,3,2
620 RESTORE 610 : FOR X%=1 TO 8 : READ N
% : Q%(X%,1)=N% : Q%(X%,8)=N% : NEXT X%
630 FOR Y%=2 TO 7 : FOR X%=1 TO 8 : IF Y
%=2 OR Y%=7 THEN Q%(X%,Y%)=1 ELSE Q%(X%,
Y%)=0
640 NEXT X% : NEXT Y% : CLS
```

re una x, il secondo un numero che si riferisce al pezzo da collocare in una data casa, 0 = nessun pezzo, 1 = Pedone, 2 = Torre, 3 = Cavallo, 4 = Alfiere, 5 = Dama, 6 = Re; il terzo e il quarto carattere sono le coordinate della casa dove va posto il pezzo. Se, a esempio, il programma incontra la mossa "d7d8x5d8", interpreta i primi quattro caratteri come una mossa normale e sposta il Pedone, perché di Pedone si deve trattare, dalla casa d7 alla casa d8 e poi inserisce il pezzo col numero 5, cioè la Dama, nella casa d8 eseguendo di fatto la promozione. Invece la mossa "d5e6x0e5" costituisce una presa al passaggio, poiché dopo la mossa del Pedone da d5 a e6, viene liberata la casa e5 come indica la semistringa x0e5, dove lo zero significa appunto nessun pezzo.

La partita termina con una delle seguenti parole chiave, scritte questa volta con caratteri maiuscoli: "Matto", "Patta" o "Abb."

Se si vogliono cambiare le partite memorizzate nel programma è sufficiente modificare soltanto le istruzioni Data, facendo però attenzione che inizino agli stessi numeri di linea attuali, cioè la prima partita dovrà iniziare alla linea 1400, la seconda alla linea 1420, la terza alla 1450 e così via. Potrebbe anche rendersi necessario un ridimensionamento della variabile con indice M$(100) qualora le nuove partite fossere costituite da più di 50 mosse.

Principali routine del programma

50 - Chiamata dello schermo grafico, definizione dei colori e dimensionamento della matrice M$() nella quale verranno caricate le mosse di una partita.
60-110 - Dati esadecimali per creare il disegno che appare accanto al titolo nella schermata iniziale.
120-190 - Schermata iniziale con il titolo.
200-210 - Attesa del comando per procedere e pulizia dello schermo.
220-250 - Valori decimali per la definizione degli sprite che serviranno da indice per la scelta delle partite e per indicare a chi spetta la mossa.
260-300 - Definizione dei suddetti sprite.
310-530 - La routine viene eseguita due volte sotto il controllo della variabile flag FG%; la prima volta vengono letti solo i nomi di tutti i giocatori per creare l'indice delle partite; la seconda volta serve per posizionare il puntatore sui dati della partita prescelta e per leggere le mosse caricandole nella matrice M$().
540-590 - Menu per selezionare le modalità del gioco.
600-640 - Definizione di una matrice bidimensionale per memorizzare la posizione dei pezzi sulla scacchiera.
650-730 - Sceneggiatura principale con il disegno della scacchiera e dei pezzi nelle case iniziali.
740 - Attribuzione dei valori iniziali ad alcune variabili di lavoro: G specifica se la mossa spetta al bianco o al nero, I% si riferisce al numero della mossa, E% è il numero totale degli errori commessi, U% sono i tentativi esauriti per indovinare una singola mossa.
750-810 - Scrittura del numero della mossa e spostamento dell'indice sul giocatore al quale spetta.
820-900 - Routine che sostituisce l'istruzione Input per introdurre la risposta.
910-950 - Valutazione della risposta.
960-1060 - Esecuzione di una mossa sulla scacchiera.
1070 - Subroutine per la creazione delle righe che formano l'indice.
1080 - Subroutine per togliere gli sprite dallo schermo.
1090-1120 - Subroutine per pulire varie porzioni dello schermo.
1130 - Subroutine per creare una pausa.
1140-1190 - Fine del programma o ritorno all'indice.
1200-1390 - Serie di routine per disegnare i vari pezzi, pedone, torre, cavallo, eccetera o per disegnare una casa vuota.
1400-1710 - Istruzioni Data contenenti le mosse delle 12 partite celebri memorizzate. ■

```
650 LINE(2,10)-(141,181),6,B : LINE(6,14
)-(137,177),6,B
660 FOR X%=8 TO 120 STEP 16 : X1%=X1%+1
 : FOR Y%=16 TO 156 STEP 20 : Y1%=Y1%+1
670 IF (X1%+Y1%)/2=INT((X1%+Y1%)/2) THEN
 C%=5 ELSE C%=4
680 LINE(X%,Y%)-(X%+15,Y%+19),C%,BF : NE
XT Y% : NEXT X%
690 C%=1 : Y%=16 : X%=8 : GOSUB 1240 : X
%=24 : GOSUB 1270 : X%=40 : GOSUB 1300 :
 X%=56 : GOSUB 1330 : X%=72 : GOSUB 1360
 : X%=88 : GOSUB 1300
700 X%=104 : GOSUB 1270 : X%=120 : GOSUB
 1240 : Y%=36 : FOR X%=8 TO 120 STEP 16
 : GOSUB 1210 : NEXT X%
710 C%=15 : Y%=136 : FOR X%=8 TO 120 STE
P 16 : GOSUB 1210 : NEXT X% : X%=8 : Y%=
156 : GOSUB 1240 : X%=24 : GOSUB 1270
720 X%=40 : GOSUB 1300 : X%=56 : GOSUB 1
330 : X%=72 : GOSUB 1360 : X%=88 : GOSUB
 1300 : X%=104 : GOSUB 1270 : X%=120 : G
OSUB 1240
730 PRESET(8,0) : PRINT #1,N$ : PRESET(8
,184) : PRINT #1,B$ : LINE(176,181)-(240
,149),1,B
740 G=-1 : I%=0 : E%=0 : U%=0
750 K$=INKEY$ : IF K$="" THEN 750 ELSE G
OSUB 1100
760 I%=I%+1 : G=-G : PUT SPRITE 2,(136,9
1+92*G),6
770 IF NOT I%/2=INT(I%/2) THEN NM$=STR$(
INT(I%/2)+1)+"." : NM$=RIGHT$(NM$,LEN(NM
$)-1) : GOSUB 1120 : PRESET(152,160) : P
RINT #1,NM$
780 IF M$(I%)="MATTO" OR M$(I%)="PATTA"
OR M$(I%)="ABB." THEN R$=M$(I%) : GOTO 1
140
790 IF I%<5 OR NOT S=G THEN 950
800 PUT SPRITE 4,(200,80),11 : PUT SPRIT
E 5,(200,96),11 : R$="      "
```

Seguito listato Chess training.

```
810 GOSUB 1090 : X%=192
820 PUT SPRITE 3,(X%,169),15
830 K$=INKEY$ : IF K$="" THEN 830
840 IF ASC(K$)>31 AND ASC(K$)<128 THEN 8
90
850 IF K$=CHR$(29) AND X%>192 THEN X%=X%
-8 : GOTO 820
860 IF K$=CHR$(28) AND X%<216 THEN X%=X%
+8 : GOTO 820
870 IF K$=CHR$(13) THEN 910
880 GOTO 750
890 P%=X%/8-23 : R$=LEFT$(R$,P%-1)+K$+RI
GHT$(R$,4-P%) : GOSUB 1090 : PRESET(192,
160) : PRINT #1,R$ : IF X%<216 THEN X%=X
%+8
900 GOTO 820
910 GOSUB 1100 : PRESET(184,136) : IF R$
=LEFT$(M$(I%),4) THEN  PRINT #1,"ESATTO"
 : U%=0 : GOTO 960
920 PRINT #1,"ERRATO" : GOSUB 1130 : E%=
E%+1 : R$="errori:"+STR$(E%) : GOSUB 111
0 : PRESET(160,8) : PRINT #1,R$
930 U%=U%+1 : IF U%<2 THEN  GOSUB 1100 :
 PRESET(184,136) : PRINT #1,"RIPROVA" :
R$="     " : GOTO 810
940 U%=0 : GOSUB 1080 : GOSUB 1100
950 GOSUB 1090 : PRESET(192,160) : R$=LE
FT$(M$(I%),4) : PRINT #1,R$
960 GOSUB 1080 : X%=ASC(LEFT$(R$,1))-96
 : Y%=VAL(MID$(R$,2,1)) : C%=5+((X%+Y%)/2
=INT((X%+Y%)/2)) : P%=Q%(X%,Y%) : Q%(X%,
Y%)=0 : GOSUB 1390
970 X%=ASC(MID$(R$,3,1))-96 : Y%=VAL(RIG
HT$(R$,1)) : C%=5+((X%+Y%)/2=INT((X%+Y%)
/2)) : Q%(X%,Y%)=P% : GOSUB 1390 : IF G=
1 THEN C%=15 ELSE C%=1
980 ON P%+1 GOTO 1050,990,1000,1010,1020
,1030,1040
990 GOSUB 1210 : GOTO 1050
1000 GOSUB 1240 : GOTO 1050
1010 GOSUB 1270 : GOTO 1050
1020 GOSUB 1300 : GOTO 1050
1030 GOSUB 1330 : GOTO 1050
1040 GOSUB 1360
1050 IF LEN(M$(I%))=4 THEN 750 ELSE M$(I
%)=RIGHT$(M$(I%),4) : R$=M$(I%)
1060 IF NOT LEFT$(R$,1)="x" THEN 960 ELS
E P%=VAL(MID$(R$,2,1)) : GOTO 970
1070 READ B$,N$ : R$=B$+" - "+N$ : PRESE
T(32,Y%) : PRINT #1,R$ : Y%=Y%+8 : RETUR
N
1080 PUT SPRITE 4,(-32,209) : PUT SPRITE
 5,(-32,209) : PUT SPRITE 3,(-32,209) :
RETURN
1090 LINE(191,159)-(224,168),4,BF : RETU
RN
1100 LINE(176,135)-(240,144),4,BF : RETU
RN
1110 LINE(215,7)-(255,16),4,BF : RETURN
1120 LINE(152,159)-(175,168),4,BF : RETU
RN
1130 FOR T%=1 TO 1000 : NEXT T% : RETURN
1140 GOSUB 1090 : PRESET(184,160) : PRIN
T #1,R$
1150 K$=INKEY$ : IF K$="" THEN 1150
1160 PUT SPRITE 2,(-32,209) : CLS : PRES
ET(8,8) : PRINT #1,"VUOI CONTINUARE? (Y/
N)"
1170 K$=INKEY$ : IF K$="" THEN 1170
1180 IF K$="Y" OR K$="y" THEN CLS : GOTO
 310
1190 IF K$="N" OR K$="n" THEN END ELSE 1
170
1200 REM pedone
1210 DRAW "C=C%;" : DRAW "BM=X%;,=Y%;" :
 DRAW "BM+7,+3NR1G1NR3D1NR3F1NR1D1NL2NR3
D1R1F3D2G1L5H1U2E3" : PAINT(X%+6,Y%+12),
C%
1220 PRESET(X%+4,Y%+16) : DRAW "C=C%;" :
 DRAW "R7F1L9" : RETURN
1230 REM TORRE
1240 DRAW "C=C%;" : DRAW "BM=X%;,=Y%;" :
 DRAW "BM+3,+2R1D1R3U1R1D1R3U1R1D1G1L7H1
" : LINE(X%+4,Y%+6)-(X%+11,Y%+12),C%,BF
1250 LINE(X%+3,Y%+13)-(X%+12,Y%+13),C% :
 PRESET(X%+3,Y%+15) : DRAW "C=C%;" : DRA
W "R9F1D1L11U1R10" : RETURN
1260 REM CAVALLO
1270 DRAW "C=C%;" : DRAW "BM=X%;,=Y%;" :
 DRAW "BM+1,+8E3D1R2U2L2E1R1U2F1D1F1NU3R
1F2D1F1D6L7U1E2U1E2H1G2L2D2H1G1H1U1" : P
AINT(X%+10,Y%+12),C%
1280 PRESET(X%+9,Y%+2) : DRAW "C=C%;" :
DRAW "F4D1F1D6L1BM-8,+2NR7G1R9" : RETURN
1290 REM ALFIERE
1300 DRAW "C=C%;" : DRAW "BM=X%;,=Y%;" :
 DRAW "BM+7,+2R1D1L1G1R3F1L5F1L1G1ND3R1D
4R1U2F1D1NR3E1NR3E1R2U2G1U2L1"
1310 DRAW "BM-4,+7R5D1L2D1F1R4F1L5H1L1G1
L5E1R4E1U1L2" : RETURN
1320 REM DAMA
1330 DRAW "C=C%;" : DRAW "BM=X%;,=Y%;" :
 DRAW "BM+1,+4R1D1F1E1U2R1D2R2U3R1D3R2U2
R1D3R1E1U1R1D1G2D1L1D3L7U3L1U1H2" : PAIN
T(X%+6,Y%+8),C%
1340 LINE(X%+4,Y%+13)-(X%+11,Y%+17),C%,B
F : RETURN
1350 REM RE
1360 DRAW "C=C%;" : DRAW "BM=X%;,=Y%;" :
 DRAW "BM+7,+2R1F1L3D1R3G1L1D1R1D1R1E2R1
F2D3L1D2L1D2L9U2L1U2L1U3E2R1F2R1F1R1E2R1
F1D2G1D1G1D1L3U4H1L1H2L1G1D2F1D1F1D1R3U4
"
1370 DRAW "C=C%;" : DRAW "BM-4,+7R9D1L9"
 : RETURN
1380 REM CASA VUOTA
```

MSX

Seguito listato Chess training.

```
1390 X%=(X%*2-1)*8 : Y%=176-Y%*20 : LINE
(X%,Y%)-(X%+15,Y%+19),C%,BF : RETURN
1400 DATA Anderssen,Kieseritzki,e2e4,e7e
5,f2f4,e5f4,f1c4,b7b5,c4b5,d8h4,e1f1;g8f
6,g1f3,h4h6,d2d3,f6h5,f3h4,h6g5,h4f5,c7c
6,h1g1,c6b5,g2g4,h5f6
1410 DATA h2h4,g5g6,h4h5,g6g5,d1f3,f6g8,
c1f4,g5f6,b1c3,f8c5,c3d5,f6b2,f4d6,b2a1,
f1e2,c5g1,e4e5,b8a6,f5g7,e8d8,f3f6,g8f6,
d6e7,MATTO
1420 DATA Spasskij,Fischer,e2e4,e7e5,f2f
4,e5f4,g1f3,g7g5,h2h4,g5g4,f3e5,g8f6,d2d
4,d7d6,e5d3,f6e4,c1f4,f8g7,b1c3,e4c3,b2c
3,c7c5,f1e2,c5d4
1430 DATA e1g1h1f1,b8c6,e2g4,e8g8h8f8,g4
c8,a8c8,d1g4,f7f5,g4g3,d4c3,a1e1,g8h8,g1
h1,f8g8,f4d6,g7f8,d6e5,c6e5,g3e5,g8g7,f1
f5,d8h4,h1g1
1440 DATA h4g4,f5f2,f8e7,e1e4,g4g5,e5d4,
c8f8,e4e5,f8d8,d4e4,g5h4,f2f4,ABB.
1450 DATA Fischer,Koronoj,e2e4,e7e5,g1f3
,b8c6,f1b5,a7a6,b5a4,g8f6,e1g1h1f1,f8e7,
f1e1,b7b5,a4b3,e8g8h8f8,c2c3,d7d6,d2d4,c
8g4,c1e3,e5d4,c3d4,c6a5
1460 DATA b3c2,a5c4,e3c1,c7c5,b2b3,c4a5,
d4d5,f6d7,b1d2,e7f6,a1b1,c5c4,h2h3,g4f3,
d2f3,c4b3,a2b3,d8c7,c1e3,f6c3,e1e2,b5b4,
f3d4,f8e8,d4f5,a5b7
1470 DATA e3d4,g7g6,f5h6,g8f8,b1c1,a8c8,
c2d3,c7a5,e2c2,d7e5,d3f1,b7c5,d4c3,b4c3,
c2c3,f8g7,h6g4,e5g4,d1g4,c8b8,c3f3,c5e4,
g4f4,f7f5
1480 DATA f3e3,e8e5,c1c6,b8e8,c6d6,a5a1,
d6a6,a1d4,e3d3,d4b2,d5d6,g6g5,f4e3,f5f4,
e3a7,ABB.
1490 DATA Keres,Szabo,e2e4,c7c5,g1f3,b8c
6,d2d4,c5d4,f3d4,g8f6,b1c3,d7d6,c1g5,e7e
6,d1d2,f8e7,e1c1a1d1,e8g8h8f8,f2f4,a7a6,
e4e5,d6e5,d4c6,b7c6
1500 DATA f4e5,f6d7,h2h4,a8b8,d2e3,f8e8,
h1h3,d8a5,g5e7,e8e7,h3g3,e7e8,d1d7,c8d7,
f1d3,h7h6,e3f4,g8f8,g3g7,f8g7,f4f6,g7f8,
d3g6,ABB.
1510 DATA Tal,Darga,e2e4,c7c5,g1f3,e7e6,
d2d4,c5d4,f3d4,g8f6,b1c3,d7d6,g2g3,a7a6,
f1g2,d8c7,e1g1h1f1,f8e7,f2f4,b8c6,d4c6,b
7c6,e4e5,d6e5,f4e5,f6d7
1520 DATA c1f4,e8g8h8f8,c3e4,c8b7,d1h5,g
7g6,h5e2,c6c5,e4f6,e7÷6,e5f6,e6e5,f4e5,d
7e5,g2b7,a8b8,b7d5,b8b6,a1e1,e5d7,e2e7,c
5c4,g1h1,b6b2
1530 DATA e1e6,b2b8,e6d6,b8d8,f1d1,ABB.
1540 DATA Cigorin,Steinitz,e2e4,e7e5,g1f
3,b8c6,f1c4,f8c5,b2b4,c5b4,c2c3,b4a5,e1g
1h1f1,d7d6,d2d4,c8d7,d1b3,d8f6,d4e5,d6e5
,f1d1,h7h6,c1a3,a8d8
1550 DATA b1d2,a5b6,c4d5,c6a5,b3b4,c7c5,
b4b2,g8e7,d2b3,a5b3,b2b3,e8g8h8f8,d5b7,e
7g6,c3c4,g6f4,b3e3,d7g4,b7d5,f8e8,a3b2,d
8d6,d1d2,f4g2
1560 DATA g1g2,g4f3,e3f3,f6g5,g2h1,g5d2,
f3f7,g8h7,a1g1,ABB.
1570 DATA Estrin,Nun,e2e4,e7e5,g1f3,b8c6
,f1c4,g8f6,f3g5,f8c5,g5f7,c5f2,e1f1,d8e7
,f7h8,d7d5,e4d5,c6d4,h2h3,f2g3,c2c3,d4f5
,d2d4,c8d7
1580 DATA d1e2,g3h2,f1f2,f5g3,e2e1,f6e4,
f2f3,e4d2,b1d2,e7f6,ABB.
1590 DATA Keres,Alechin,e2e4,e7e5,g1f3,b
8c6,f1b5,a7a6,b5a4,d7d6,c2c4,c8d7,b1c3,g
7g6,d2d4,f8g7,c1e3,g8f6,d4e5,d6e5,e3c5,f
6h5,c3d5,h5f4
1600 DATA d5f4,e5f4,e4e5,g6g5,d1d5,g7f8,
c5f8,h8f8,e1c1a1d1,d8e7,a4c6,d7c6,d5d3,c
6d7,f3g5,e8c8a8d8,g5f3,f7f6,e5f6,f8f6,h1
e1,e7b4,d3d7,ABB.
1610 DATA Tal,Letelier,e2e4,e7e5,g1f3,b8
c6,f1b5,a7a6,b5a4,d7d6,c2c3,c8d7,d2d4,g8
e7,a4b3,h7h6,f3h4,g7g5,d1h5,h8h7,c1g5,e5
d4,f2f4,d8c8,f4f5,d4c3
1620 DATA b1c3,c6e5,g5f6,e7g8,f6e5,d6e5,
h4g6,f8d6,b3f7,e8f7,c3d5,f7g7,e1g1h1f1,g
8f6,d5f6,g7f6,g6e5,c8e8,e5d7,h7d7,e4e5,d
6e5,h5h6,f6f7
1630 DATA a1e1,d7d5,h6h7,f7f6,e1e4,e5d4,
g1h1,ABB.
1640 DATA Lasker,Alechin,e2e4,d7d5,e4d5,
g8f6,d2d4,f6d5,g1f3,c8g4,c2c4,d5b6,b1c3,
e7e5,c4c5,e5d4,c3e4,b6d7,d1d4,d8e7,f1b5,
b8c6,b5c6,b7c6
1650 DATA e1g1h1f1,g4f3,g2f3,e8c8a8d8,d4
a4,d7e5,g1g2,e7e6,a4a7,e6f5,a7a8,c8d7,f1
d1,d7e6,a8d8,f5f3,g2g1,f8e7,d8d4,f3g4,g1
h1,g4f3,PATTA
1660 DATA Spielmann,Keres,e2e4,e7e6,d2d4
,d7d5,b1c3,g8f6,e4e5,f6d7,f2f4,c7c5,d4c5
,b8c6,a2a3,f8c5,d1g4,g7g6,g1f3,a7a6,f1d3
,b7b5,b2b4,c5a7
1670 DATA h2h4,h7h5,g4g3,d8e7,f4f5,a7b8,
f5g6,d7e5,g6f7,e7f7,f3g5,f7f6,h1f1,e5f4,
g3f3,f6c3,e1d1,c3g7,f3e2,h8f8,f1f8,e8f8,
g5e6,c8e6
1680 DATA e2e6,g4f2,d1e1,f2d3,ABB.
1690 DATA Smyslov,Donner,e2e4,e7e6,d2d4,
d7d5,b1c3,g8f6,c1g5,f8b4,e4e5,h7h6,g5d2,
b4c3,b2c3,f6e4,d1g4,g7g6,d2c1,e4c3,f1d3,
c7c5,d4c5,b8c6,g1f3
1700 DATA d8a5,e1g1h1f1,a5c5,g4h4,c6e7,h
4f6,h8h7,a2a4,c5c7,c1a3,c3e4,d3e4,d5e4,f
3d2,e7d5,f6h4,c7e5,d2e4,g6g5,h4g4,c8d7,e
4d6,e8d8
1710 DATA d6b7,d8c8,b7d6,c8c7,g4c4,d7c6,
c4c5,e5f4,c2c4,d5b6,a4a5,ABB.
```

Input simulato

Un'utile routine per risolvere i problemi degli input numerici e alfanumerici

di Maurizio Galluzzo

L'utilizzo del comando Basic Input, comporta nello standard Msx alcune limitazioni. È possibile durante un Input numerico inserire dei caratteri non desiderati e per questo incorrere in errore con ripetizione dell'Input stesso e provocare delle scritte indesiderate nello schermo (Redo from start). L'Input alfanumerico presenta altri inconvenienti. Provate a scrivere questo esempio:

```
10 INPUT A$
20 PRINT A$
30 GOTO 10
```

Il programmino vi chiederà continuamente una stringa ma, se voi inserite degli spazi e quindi dei caratteri, la stampa avverrà tralasciando completamente gli spazi e creando spesso problemi di impaginatura. Provate ora a scrivere la seguente frase: "abcde,fgh" alla richiesta di Input dal precedente programma.

Anche in questo caso sullo schermo appare una segnalazione: "Extra ignored" e verranno stampati i soli caratteri prima della virgola. Questo perché con il comando Input si possono chiedere più dati, separandoli dalla virgola. Questi casi dovrebbero avervi fatto capire come non si possa fare un uso diretto dell'istruzione Input in programmi destinati a terzi (programmi commerciali) specie se gli input in questione avvengono all'interno di "cartelle" cioè schermi che simulano la presenza di un documento.

Il programma presentato assolve alle funzioni di Input sia numerico che alfanumerico. Nell'Input numerico vengono accettati i numeri (0 - 9) e il punto o la virgola. Nei programmi commerciali è bene prevedere entrambe le possibilità in quanto il software può essere destinato a utenti non molto abituati al sistema di notazione decimale anglosassone. L'Input alfanumerico accetta invece tutti i caratteri con codice maggiore o uguale a 32 (space), i caratteri di BS (cancellazione ultimo carattere introdotto) e di Return.

È da notare che nell'Input numerico i caratteri diversi da quelli descritti non vengono accettati, compresi anche i punti superiori al primo. La stringa può essere corretta in questo caso con BS. Questa funzione tiene conto della cancellatura del punto e ne permette quindi la sostituzione.

Per usare il programma autonomamente è necessario inizializzare le variabili x e y che indicano la posizione dove deve essere scritta la stringa e quindi dare Goto 250 per la stringa alfanumerica e Goto 90 per l'Input numerico. Alla riga 910 il carattere 219 indica il cursore, ma può venire sostituito con un altro modificando il numero tra parentesi.

Negli Input stringa non si possono comunque superare i 255 caratteri (è la limitazione prevista dall'Msx Basic).

Commento alle singole istruzioni

10-240 - Inizializzazione del programma e delle variabili di posizionamento.

250-400 - Input alfanumerico / accettazione e controllo dei dati.

410-480 - Stampa stringa.

490-820 - Input numerico / posizionamento.

830-950 - Stampa stringa

960 1070 - Stampa stringa finale senza il cursore.

Elenco variabili

F$ - Stringa totale

FL - Flag punto

H$ - Stringa parziale

H - Valore numerico Input

J$ - Stringa in Input

J - Codice carattere

L - Lunghezza stringa

X - Coordinata di stampa

Y - Coordinata di stampa.

Listato 1 - Il programma Input simulato.

```
10 REM ****************************
20 REM *                          *
30 REM *     INPUT SIMULATO       *
40 REM *                          *
50 REM *                          *
60 REM * di Maurizio Galluzzo     *
70 REM *                          *
80 REM ****************************
90 REM
100 REM
110 CLEAR 500
120 SCREEN 0
130 KEY OFF
140 CLS
150 LOCATE 6,2
160 PRINT "Esempio di INPUT simulato"
170 '
180 '
190 X=20 : '    valore da modificoare
200 Y=10 : '    valore da modificare
```

Seguito listato Input simulato.

```
210 LOCATE 0,Y
220 PRINT "Input alfanumerico"
230 '
240 '
250 REM
260 REM *************************
270 REM *                       *
280 REM *   INPUT ALFANUMERICO  *
290 REM *                       *
300 REM *************************
310 REM
320 H$=""
330 GOSUB 830
340 J$=INKEY$
350 IF J$="" THEN 340
360 J=ASC(J$)
370 IF J>=32 THEN H$=H$+J$:GOSUB 830
380 IF J=8 AND H$<>"" THEN L=LEN(H$):H$=
    MID$(H$,1,L-1):GOSUB 830:GOTO 340
390 IF J=13 THEN GOSUB 960:GOTO 410
400 GOTO 340
410 REM
420 REM *************************
430 REM *                       *
440 REM *     STAMPA STRINGA    *
450 REM *                       *
460 REM *************************
470 REM
480 PRINT:PRINT:PRINT H$
490 REM
500 REM *************************
510 REM *                       *
520 REM *     INPUT NUMERICO    *
530 REM *                       *
540 REM *************************
550 REM
560 '
570 '
580 X=20 : '   valore da modificare
590 Y=15 : '   valore da modificare
600 LOCATE 0,Y
610 PRINT "Input numerico"
620 '
630 '
640 FL=0
650 H$=""
660 GOSUB 830
670 J$=INKEY$
680 IF J$="" THEN 670
690 J=ASC(J$)
700 IF J=13 THEN GOSUB 960:GOTO 790
710 IF J=8 AND H$<>"" THEN L=LEN(H$):K$=
    MID$(H$,L,L):IF K$="." THEN FL=0
720 IF J=8 AND H$<>"" THEN L=LEN(H$):H$=
    MID$(H$,1,L-1):GOSUB 830:GOTO 670
730 IF FL=0 AND J$="," OR FL=0 AND
    J$="." THEN FL=1:J$=".":GOTO 760
740 IF J$<"0" OR J$>"9" THEN 670
750 IF J$="." THEN 670
760 H$=H$+J$
770 GOSUB 830
780 GOTO 670
790 REM calcolo numerico
800 H=VAL(H$)
810 PRINT:PRINT:PRINT H
820 END
830 REM
840 REM *************************
850 REM *                       *
860 REM *     STAMPA STRINGA    *
870 REM *                       *
880 REM *************************
890 REM
900 BEEP
910 F$=H$+CHR$(219)+CHR$(32)
920 LOCATE X,Y
930 PRINT F$
940 F$=""
950 RETURN
960 REM
970 REM *************************
980 REM *                       *
990 REM *   STAMPA DEFINITIVA   *
1000 REM *                      *
1010 REM *************************
1020 REM
1030 BEEP
1040 F$=H$+CHR$(32)
1050 LOCATE X,Y
1060 PRINT F$
1070 RETURN
```

Bulletin board: la nuova invasione

Con il 1° Ottobre 1985 ha iniziato ufficialmente a operare a Firenze la prima banca dati completamente gratuita a disposizione degli operatori del settore e degli appassionati di home e personal computer.
Il sistema realizzato da un giovane softwarista fiorentino, Giorgio Verzoletto, ha dimostrato già dai primissimi giorni di attività di poter raccogliere vasti consensi.
Mail box, bulletin board, archivio software, elenco emittenti radiofoniche, spettacoli, giuochi, sono solo alcuni degli innumerevoli servizi che la Italdata Service offre ai propri utenti.
Per chi volesse provare il collegamento i parametri sono: 300 baud, full duplex, nessuna parità, 7 bit.
Stabilito il contatto con la Italdata Service digitando il codice "Logom", si può accedere alle diverse opzioni previste. Per molte di queste è necessario essere in possesso di una casella e di un codice riservato. Anche questa operazione può essere effettuata direttamente dal terminale ed è, come tutti gli altri servizi, completamente gratuita.

Italdata Service
Tel. 055-474680
50100 Firenze

Concorso Call for Games '86

Quest'anno *Call for Games* si fa in grande stile, con la collaborazione delle principali riviste nazionli (inclusa **Personal Software**, naturalmente). Call for Games, per chi ancora non lo sapesse, è un concorso aperto a tutti gli autori di "software interessante", giunto ormai alla sua quarta edizione.
"Software interessante" vuol dire programmi per giochi di ogni tipo, didattica, e anche intelligenza artificiale (che so, un robot che scriva giochi per vincere questo concorso...).
Call for Games è organizzato dall'Aica (Associazione italiana di calcolo automatico), che lavora tra l'altro per la promozione della cultura informatica a tutti i livelli.

Se qualcuno ancora ritiene i giochi per computer estranei alla cultura informatica, vada a sentire "Computer Play '86", che si svolgerà in Settembre in coincidenza con il Sim '86 (salone della musica, dell'elettronica di consumo e dell'informatica casalinga).

Avrà così anche l'occasione di assistere alla premiazione dei vincitori di Call for Games, tra i quali ci potreste essere anche voi che state leggendo!

Partecipare? Semplicissimo. Scrivete un bel programma, o tiratelo fuori dal cassetto dove l'aveto tenuto in attesa della grande occasione, e mandatelo alla redazione di **Personal Software** completo di:

- il programma completo e funzionante su dischetto o cassetta;
- il listato di tutte le parti del programma, su carta (se non trovate una stampante, mandatelo su dischetto o cassetta). I diritti d'autore rimangono comunque vostri;
- la descrizione di quello che il programma fa e di come si usa. Ci sono premi per varie categorie (animazione, avventure, grafica, musica ...), comunque non è essenziale che classifichiate il programma in una di queste, a meno che non vogliate sottolineare un aspetto del vostro lavoro che ritenete particolarmente significativo;
- chi siete, cosa fate, come vi è venuta l'idea di scrivere questo programma, cos'ha di interessante o di originale dal punto di vista dell'uso o della realizzazione, eccetera;
- non dimenticate: nome, indirizzo e numero di telefono! (succede anche questo...);
- il tutto entro Luglio 1986 (non aspettate l'ultimo minuto).

E poi, che succede? Faremo, qui in redazione, una preselezione dei programmi più validi, che saranno iscritti a partecipare a Call for Games '86. Alcuni di questi saranno anche pubblicati su **Personal Software**, senza comunque che questo influenzi la giuria dell'Aica (un programma interessante arrivato all'ultimo minuto non può ovviamente essere pubblicato).
Se il vostro programma sarà classificato, vi aspettano la gloria e un'ardua prova: dovrete spiegare alla Giuria, presente il pubblico di Computer Play '86, le caratteristiche del programma, le soluzioni tecniche e di regia, eccetera (riceverete maggiori dettagli in seguito; per ora, vedete di classificarvi).

Un consiglio: cercate di fare qualcosa di originale.

È molto improbabile farsi notare con il milionesimo Mastermind, per quanto ben fatto.

E curate i dettagli: sono sempre importanti.
Forza, che aspettate? Datevi da fare: vogliamo dire che i vincitori di Call for Games '86 sono stati presentati da **Personal Software**. ■

PERSONAL MARKET

Commodore

Vendo **interfaccia per collegare due registratori** al C 64 o al VIC 20 a L. 100.000. Alessandro Guarino - Via Gorizia, 9 - 36010 Cavazzale (VI)

Vendo **SpeeDOS** per C 64 con manuale a L. 120.000. Velocizza di 10 volte in lettura e scrittura. Telefonare dopo le 20.00. Eliseo Mastrangelo - Via Casilina, 1641 - 00133 Roma - Tel. 06/6151345

Vendo **VIC 20 in ottime condizioni** con 3 cassette giochi e 3 cartridge a L. 150.000. Affrettatevi! Matteo Recanatini - Residenza Spiga, Scala 9 - 20090 Milano 2-Segrate (MI) - Tel. 02/2141528

Vendo **C 64 più stampante MPS 802,** drive 1541, SpeeDOS, monitor a fosfori verdi, 50 programmi a scelta fra 3.000 a L. 1.200.000. Oppure cambio con SX 64 Executive. Massimo Venturini - Cannareggio, 96E - 30121 Venezia - Tel. 041/714283

Vendo **Commodore 8000 con doppio floppy da 500 + 500 Kbyte,** video, stampante veloce a 132 colonne, completo di programmi gestionali. Tutto come nuovo, prezzo affare. Claudio Giovanelli - Via Ripamonti, 194 - 20141 Milano - Tel. 02/536926

Vendo **C 16 più registratore,** 10 cassette originali, joystick, libro "Introduzione al BASIC" parte prima. Pietro Ghisu - Via G. Oceano, 12 - 08100 Nuoro

Vendo **printer plotter Commodore 1520** in ottime condizioni con imballo, carta pennini e software home made. Tutto a L. 180.000. Telefonare ore 19.30-21.30. Federico Scaramuzza - Via Divisione Torino, 29 - 00143 Roma - Tel.06/5916360

Per C 64 vendo a L. 30.000 **eccezionale programma "Oracolo" su disco.** Avrai una sicura risposta a qualsiasi domanda sul tuo destino. Annuncio sempre valido. Rivolgersi a: Salvatore Daidone - Via I. Lampiasi, 20 - 91100 Trapani

Eccezionale! Vendo **C 64 più registratore C2N,** floppy 1541, tavoletta Koala, joystick, SpeeDOS Plus, 30 cassette, 30 manuali, 130 dischetti e 1.000 programmi. Per informazioni rivolgersi a: Jacopo Chiappetti - Via Vertoiba, 7 - 20137 Milano - Tel. 02/571608

Per C 64 vendo in blocco, a L. 25.000, **9 programmi Totocalcio,** tra i quali Tot 13, Toto Professionisti, Suprridotti, Toto Plus. Massima serietà. Francesco Cappuccio - Via N. Sauro, 3 - 8100 Caserta - Tel. 0823/324403

Per C 64 vendo **Simons' BASIC originale Commodore** più manuale a L. 20.000. Perfettamente funzionante. Giampietro Ceci - Via Eliano, 28 - 00036 Palestrina (RM) - Tel. 06/9557698

Cambio **cartridge e programmi espansi** per VIC 20. Cinzia Ceccarini - Via Di Vittorio, 10 - 58022 Follonica (GR)

Vendo **C 64 e registratore datassette** più manuale d'uso, giochi, programmi e libri vari; il tutto, in ottimo stato e con imballi originali, a L. 390.000. Stefano Longhi - Via Verdi, 25 - 20053 Muggiò (MI) - Tel. 039/793390

Cambio **software** per C 64. Dispongo di molti programmi di utility, radioamatori, elettronica, gestionali, ingegneria e numerosi manuali. Massimo Cantelli - Via Corso, 40 - 40051 Altedo (BO) - Tel. 051/871270

Sinclair

Occasione! Vendo **ZX81 più espansione 16 Kbyte,** completo di manuali e in confezione originale. Per le offerte telefonare a: Vincenzo Musicò - Via Paolo Blandino, 12 - 98100 Messina - Tel. 090/2938626

Vendo **ZX Spectrum 48 Kbyte più interface 1,** microdrive, stampante Alphacom 32, penna ottica, libro della J.C.E. "Sinclair ZX Spectrum Assembler e linguaggio macchina per principianti" completo di cassetta originale. Il tutto ancora imballato e completo di manuali originali a sole L. 669.000. Lorenzo Moro - Via S. Michele, 50 - 56100 Pisa

The A.D. Spectrum Club **cerca soci in tutta la Sicilia** per scambio e produzione software. Per informazioni e iscrizioni telefonare a: Daniele Russo - Via Genova, 57 - 95127 Catania - Tel. 095/383543

Vendo **ZX Spectrum 48 Kbyte più manuali,** cavetti, box sonoro, interfaccia joystick con tre porte, di cui una per cartucce originali e non, a L. 350.000. Telefonare dopo le 20.00. Francesco Messina - Via Teramo, 30 - 20142 Milano - Tel. 02/816128

Vendo **Spectrum 48 Kbyte completo** più ZX printer, 2 manuali CPU, libro "77 programmi", varie cassette, 20 riviste e Turbo Tape da 3.600 baud a L. 400.000. Telefonare ore 16.00-20.00. Vincenzo Romano - Via Sannita I Trav. SX, 38 - 81020 Cervino (CE) - Tel. 0823/41152

Per Spectrum 48 Kbyte vendo **programma Supertoto** per lo sviluppo e riduzione dei sistemi Totocalcio, scritto in LM, a L. 30.000. Francesco Arculeo - Via A. Costa, 6 - 90125 Palermo - Tel. 091/444432

Causa passaggio a sistema superiore vendo **periferiche** per Spectrum. Floppy, joystick e printer a prezzi da concordare. Telefonare o scrivere a: Igor Bonat - Via Ressmann, 6 - 34141 Trieste - Tel. 040/76871

Vendo a L. 20.000 **libro "ZX Spectrum - Linguaggio macchina avanzato" nuovissimo** e con cassetta di routine allegata (valore commerciale L. 30.000). Marco Sivori - Via Barchetta, 18/9 - 16162 Bolzaneto (GE) - Tel. 010/ 403118

Vendo a L. 600.000 **Spectrum Plus 48 Kbyte** più registratore Philips da sistemare, un altro registratore funzionante, joystick con interfaccia Kempston, 300 programmi di vario tipo, riviste, manuali. Claudio Bergonzoni - Via Fioravanti, 7 - 40128 Bologna - Tel. 051/358925

Vendo **ZX Spectrum più interface 1, microdrive,**, stampante Sinclair, penna ottica, interfaccia joystick, 8 cartridge, 54 cassette (anche C-90). Il tutto a L. 1.000.000 trattabile. Andrea Clivio - Via Beinotti, 12 - 16132 Genova - Tel. 010/3991084

Vendo **Spectrum 16 Kbyte mai usato,** con cavi, alimentatore, supergaranzia, manuale in italiano, 30 giochi; il tutto a L. 300.000. Massimo Corradi - Via L. Cozzani, 21 - 19100 La Spezia - Tel. 0187-706879

Vendo **ZX Spectrum 48 Kbyte più registratore Pooh,** 50 cassette e 3 libri a L. 350.000. Inoltre vendo stampante Alphacom 32 a L. 100.000 e interfaccia a 2 porte programmabile Tenkolek più 2 joystick a L. 100.000. Cristian Disanto - Via Grandi, 2 - 20135 Milano - Tel. 02/737044

Strabiliante, per sole L. 5.000, pari al prezzo dell'iscrizione, il Club Spectromania offre **15 programmi ad altissima risoluzione.** Rivolgersi per informazioni a: Club Spectromania - Via delle Romite, 8 - 50124 Galluzzo (FI) - Tel. 055/2048905

Vendo **Spectrum 48 Kbyte in garanzia** a L. 250.000 con software. Inoltre vendo stampante Seikosha GP50S nuova a L. 250.000 e interface 1 più microdrive (ancora imballati) a L. 200.000. Roberto Azzimonti - Via Ronchetto, 9 - 21050 Busto Arsizio (VA) - Tel. 0331/626570

Cambio **idee e programmi** per QL Sinclair e il nuovo Atari ST. Rispondo rapidamente a tutti. Roberto Ghezzi - Via Volontari del Sangue, 202 - 20099 Sesto San Giovanni (MI) - Tel. 02/2485511

Acquisto per Spectrum i programmi **Music Maker e Music Master.** Daniele Bricchi - Milano - Tel. 02/563895

Texas

Vendo **TI 99/4A completo più Extended BASIC,** memory 32 Kbyte, 4 SSS (Parsec, Scacchi, Tombstone City, Minus Mission), 11 Newsoft, 3 libri, 5 cassette di programmi. Antonio Casella - Via Montiglioni, 58 - 00046 Grottaferrata (RM) - Tel. 06/8456527

Vendo **TI 99/4A nuovo** completo di cavi, manuali, cassetta Beginner's BASIC Tutor e modulo SSS Personal Data base. Prezzo da concordare. Renzo Clari - Via Duse, 14 - 34170 Gorizia - Tel. 0481/81655

Vendo **TI 99/4A più Extended BASIC** con manuali in italiano, sintetizzatore vocale, SSS Pac Man, Connect Four, giochi su cassetta, libro a L. 450.000! Telefonare dalle ore 12.30 alle 14.00. Fabio Maurano - Via F. Crispi, 20 - 22100 Como - Tel. 031/273841

Vendo **nuova simulazione di volo in tempo reale** in Extended BASIC per TI 99/4A a L. 7.000. Possiede 10 strumenti (4 analogici) e si vede il terreno. Migliorato rispetto alla prima versione. Aldo Tomaselli - Via Roma, 20 - 38059 Strigno (TN) - Tel. 0461/762007

Per TI 99/4A vendo **modulo Editor Assembler** a L. 100.000. Telefonare ore 20.00-21.00. Sebastiano Tomasello - Milano - Tel. 02/270759

Vendo **TI 99 più peripheral box** con espansione RAM, RS-232, disk controller e disk drive, stampante (Epson) e molti programmi fra cui il TI-Writer, Extended BASIC, Parsec, ecc. Alessandro Carini - Via Belvedere, 15 - 22070 Figliaro (CO) - Tel. 031/940428

Compro **riviste 99er Magazine** per TI 99/4A anche in fotocopia. Precisare quantità e prezzi. Inoltre cerco mini memory ed Extended BASIC sempre per il TI. Mettersi in contatto epistolare. Gian Luigi Calzolaro - Via M. della Benedicta, 3/12 - 16010 Rossiglione (GE)

Vendo **libri riguardanti il TI 99/4A** italiani e stranieri a prezzi modici, causa passaggio a sistema superiore. Maurizio Ortolani - Via Dante Alighieri, 41 - 61100 Pesaro

Cerco **utente Texas interessato all'apprendimento dell'Assembly Language** con il quale scambiare informazioni e conoscenze. Andrea Caracciolo - Viale Piave, 8 - 20129 Milano - Tel. 02/793201

Vendo per TI 99/4A **i seguenti moduli SSS:** Extended BASIC L. 150.000; Buck Rogers L. 35.000; Donkey Kong L. 25.000; Beginning Grammar L. 15.000; Physical Fitness L. 15.000. Giovanni Alarici - Via A. Costa, 8 - 41051 Castelnuovo Rangone (MO) - Tel. 059/536477

Vendo **TI 99/4A con modulo SSS Extended BASIC** e manuali d'uso. In omaggio numerosi libri, riviste e programmi in cassetta. Giovanni Malkowski - Via Bottego, 8 - 48015 Cervia (RA) - Tel. 0544/970435

Vendo **TI 99/4A più registratore,** peripheral box, drive, 32 Kbyte RAM, Speech, minimemory, Logo, Editor/Assembler, Extended BASIC. Tutto, in imballi originali e con manuali, a L. 1.600.000 trattabili. Telefonare ore pasti. Daniele Cardelli - Via Scandicci Alto, 16 - 50018 Scandicci (FI) - Tel. 055/254681

Vendo **TI 99/4A completo di tutto** a L. 200.000 più 3 moduli gioco, Extended BASIC, minimemory, Personal Record Keeping, registratore e tutti i manuali. Paolo D'Errico - Viale Cadorna, 11 - 21052 Busto Arsizio (VA) - Tel. 630541

Si vende **sistema TI 99/4A** per L. 700.000. Consolle, Extended BASIC con manuale, registratore, joystick, 4 moduli, 13 cassette TI 99 Newsoft, 7 libri e 70 riviste. Donato De Luca - Via Soave, 30 - 20135 Milano - Tel. 02/592973

Vendo **TI 99/4A più joystick,** cassetta "Impariamo il BASIC", Extended BASIC completo di manuale in inglese, libri "L'home computer TI 99/4A" e "A scuola con il TI 99/4A", manuale d'uso e 130 programmi. Tutto in ottime condizioni, L. 500.000 trattabili in blocco. Marco Riva - Via Jenner, 10 - 20159 Milano - Tel. 02/6883733

Cerco **utente Texas interessato all'apprendimento dell'Assembly Language** con il quale scambiare informazioni e conoscenze. Andrea Caracciolo - Viale Piave, 8 - 20129 Milano - Tel. 02/793201

Vendo **TI 99/4A più manuale d'uso,** modulo Extended e registratore a sole L. 600.000 in blocco. In omaggio centinaia di programmi (giochi, utility), 5 moduli SSS (The Attack, Munch Man, ecc.) e 8 cassette New Soft. Massimo Adamo - Via Damasco, 25 - 00146 Roma - Tel. 06/5410272

Vera occasione! Causa servizio militare vendo **TI 99/4A usato veramente poco** a L. 170.000 trattabili. Inclusi alimentatore, modulatore TV, cavo registratore e manuale. Telefonare ore pasti. Pasquale Principato - Via Dell'Evangelista, 16 - 06070 Perugia - Tel. 075/44358

Causa passaggio a sistema superiore vendo **TI Extended BASIC completo di manuale** a L. 150.000 più varie cassette e libri sul TI 99/4A. Scrivetemi! Maurizio Ortolani - Via Dante Alighieri, 41 - 61100 Pesaro

Compro **consolle TI 99/4A** in buone condizioni. Ercole Donati - Via Appennini, 13 - 20151 Milano - Tel. 02/353356

Vendo **TI 99/4A** più 2 joystick e 5 giochi oppure cambio con software per Spectrum 48 Kbyte. Andrea Torelli - Via Garibaldi, 12 - 46100 Mantova - Tel. 0376/360982

Vendo **TI 99/4A completo di manuali e in imballo originale** più cavo registratore, cassetta Extended BASIC, Video Chess e 4 cassette piene di giochi a sole L. 400.000. Offro come omaggio all'acquirente 15 numeri della rivista "Paper Soft". Claudio Giordano - Via Napoli, 43 - 80078 Pozzuoli (NA) - Tel. 081/8673356

Varie

Gratis qualunque programma utility o game **iscrivendoti al Crispiansoft Club.** Molti titoli a tua disposizione al solo prezzo del supporto. Per informazioni rivolgersi a: Crispiansoft Club - Box 23 - 74012 Crispiano (TA)

Cambio **programmi per computer Atari** su disco e cassetta. Originali USA e United Kingdom. Luigi Servolini - Via La Spezia, 81 - 000182 Roma - Tel. 06/7581219

Vendo **Personal Software dal n. 1 al n. 31** a L. 100.000, Microcomputer dal n. 28 al n. 44 a L. 60.000, Micro e Personal Computer dal n. 39 al n. 57 a L. 60.000. Giorgio Castagnaro - Via Buonarroti, 13 - 21013 Gallarate (VA) - Tel. 0331/796139

Vendo **PC Digital 128 Kbyte RAM,** 2 floppy 800 Kbyte, video, tastiera; stampante Digital Lago grafica, 100 cps. più LQ. Espansioni: 2 floppy 800 Kbyte o hard disk 5/10 Mbyte, 896 Kbyte RAM, grafica, colore e sistema operativo MS-DOS, CP/M. Regalo espansione 128 Kbyte. Fabio Bonfissuto - Viale Casiraghi, 104 - 20099 Sesto S. Giovanni (MI) - Tel. 02/2424103

Vendo **Atari** con i seguenti giochi: Moon Patrol, Cristal Castles, Space Invaders, Beamrider, Centipede, Mario e Bros. Il tutto a L. 350.000. Angelo Bonfanti - Via Carlo Porta, 8 - 22052 Cernusco Lombardone (CO) - Tel. 039/597209

Vendo **raccolta completa Personal Software dal n. 1 al n. 31** a L. 150.000 o cambio con modulo SSS Extended BASIC con manuale per TI 99/4A. Ermes Colombini - Via Tignale del Garda, 55/32 - 41100 Modena - Tel. 059/393436

Vendo come nuovi: **Lemon II (Apple compatibile),** monitor fosfori verdi, drive più controller da 0,5 Mbyte, manuali di sistema, 30 dischetti con tantissimo software e relativi manuali, il tutto a un prezzo eccezionale. Telefonare a Lucio, tel. 071/200452

Vendo **amplificatore Technics SU-V4A** 60+60 W RMS, casse acustiche a 3 vie 100+100 RMS (diam. woofer 30 cm) JBL 99 VX a L. 600.000. Solo casse L. 350.000. Telefonare ore pasti. Fabrizio La Rosa - Via Chiesa dei Marinai, 12 - 98100 Messina - Tel. 090/40637

Vendo causa realizzo **Sega SC 3000H** più cavetti, presa cuffia, espansione 32 Kbyte, programmi in LM e centinaia di listati a L. 370.000. Ancora in garanzia. Paolo Orione - Via Accinelli, 11/12 - 16122 Genova - Tel. 010/221040

Vendo o cambio **programmi** per MZ-700 e MZ-80. Walter Narcisi - Via Val Gardena, 41 - 63037 Porto D'Ascoli (AP) - Tel. 0735/659758

Causa inizio università vendo **C 64, Laser 2000, stampante GP-700A,** drive 1541, datassette 1530: ottimo stato e funzionamento garantiti. Regalo ad acquirenti oltre 700 programmi per C 64. Telefonare il sabato e la domenica. Sergio Innocente - Via Saccardo, 16 - 31100 Treviso - Tel. 0422/51182

Vendo **Rockwell AIM-65,** 4 Kbyte RAM più BASIC, Assembler ed interfaccia video in ottime condizioni e completo di manuali. Prezzo interessante. Gianni Botto - Via Roma, 20 - 10088 Volpiano (TO0 - tel. 011/9884745

Vendo **Seikosha GP550A** a L. 750.000; interfaccia RS-232 più cavo L. 140.000. In omaggio software per interfacciare stampante con Spectrum 48 Kbyte. Telefonare a: Riccardo - Via Spartaco, 26 - 20135 Milano - Tel. 02/5469734

Vendo **MZ-731 Sharp più 5 diversi tipi di BASIC,** Pascal Interpreter, Lisp Interpreter, Pascal, compilatori Fortran, assemblatore/disassemblatore, linguaggio macchina, 12 riviste tutto Sharp e manuali. Telefonare ore pasti. Maurizio Rizzi - Via Cavalli, 24 - 36100 Vicenza - Tel. 0444/565175

Sono un ragazzo di Milano e cerco **possessori di Sharp MZ-700** per eventuali scambi di software. Telefonate o almeno scrivete: prometto di rispondere a tutti. Francesco Loriga - Via Fulda, 115 - 00148 Roma - Tel. 06/5222504

Vendo **riviste americane "Compute!" e "Compute!'s Gazette"** a L. 3.000 a fascicolo. Vendo inoltre numeri di "Modellistica" 76-83 e "RC Modeller" 81-84. Marcello Beucci - Piazza Giotto, 13 - 52100 Arezzo - Tel. 0575/26906

Per Olivetti M20 cambio **gioco del Poker e Bobo** più programma per gestione testi, tutti realizzati da me, con compilatore Assembler sempre per M20. Antonio Moro - Via A. De Gasperi, 56 - 36022 Cassola (VI) - Tel. 0424/83027

Cambio **software per Atari 520 ST.** Cerco inoltre informazioni sulla macchina e su eventuale Club Atari 520 ST. Telefonare ore 13.00-14.00. Marco Fattorini - Via L. Viani, 21 - 50142 Firenze - Tel. 055/711629

Vendo **micro Z80 N.E. 56 Kbyte RAM,** 1 drive, scheda grafica LX529 a L. 1.400.000; monitor 12" fosfori verdi L. 200.000; Oki ML80 L. 400.000. Tratto con il Veneto. Marino Ferla - Via Liguria, 7 - 30030 Chirignago (VE) - Tel. 041/915749

Vendo **Atari 800 XL** più tavoletta grafica, registratore cassette 1010, 2 cartucce gioco, 10 cassette gioco, 2 libri Atari. Tutto a L. 400.000. Laura Termine - Via Galla e Sidama, 40 - 00199 Roma - Tel. 06/836410

Vendo **computer Adam, consolle Colecovision** con 5 giochi, word processor, Logo a prezzo trattabilissimo. Tutto in perfette condizioni. Roberto Scassellati - Via dei Fornacia, 6 - 26100 Cremona - Tel. 0372/434211

Cerco **programmi per Sharp PC-1401.** Cambio con programmi per C 64, C 16, Apple ed altri. Maurizio Priano - Via Opisso, 115/81 - 16155 Genova

Vendo **computer IBM compatibile 256 Kbyte RAM** con video e doppio floppy, completo di programmi gestionali. Claudio Giovanelli - Via Ripamonti, 194 - 20141 Milano - Tel. 02/563105

Vendo in blocco **Personal Software dal nr. 2 al nr. 19** a L. 30.000; M&P Computer dal nr. 1 al nr. 40, sempre in blocco, L. 1.500 a rivista più spese postali. Spedizione contrassegno. Regalo Informatica Oggi, Chip, ecc. Daniele Fonsato - Via Falconero-Villa Selenia - 10057 S.Ambrogio di Torino (TO)

Vendo **Sharp MZ-731;** televisore portatile a colori Philips; piastra cassette Pioneer CT-7R dotata di microprocessore. Telefonare al venerdì dopo le ore 16.00. Carlo Carollo - Via Rovigo, 16 - 32100 Bolzano - Tel. 0471/38819-933588

Vendo **Sony MSX più data base 16 Kbyte incorporato,** registratore, video 12" ambra con suono, stampante grafica, joystick, 5 manuali, mobile, programmi, garanzia a L. 1.400.000. Marco Martini - Via Don Perduca, 51/C - 15057 Tortona (AL) - Tel. 0131/867219

Vendo **raccolta 1984 di Paper Soft** e riviste (Bit, Personal Software). Vendo cassetta con 50 giochi per TI 99/4A a L. 30.000. Solo zona Milano. Andrea Casati - Via Manzoni, 15 - 20090 Trezzano S/N (MI) - Tel. 02/4452080

Cambio **software per MZ-700** e MZ-80K/A. Walter Narcisi - Via Val Gardena, 41 - 63037 Porto d'Ascoli (AP) - Tel. 0735/659758

Vendo **Sharp MZ-721** causa passaggio a sistema superiore, completo di programmi, catalogo, manuali. Prezzo: L. 500.000 trattabili. Rivolgersi a: Giacomo Monnanni - Via Mercurio, 45 - 52100 Arezzo - Tel. 0575/352423

Vendo **controllore di alimentazione** a otto prese protette dai disturbi di rete con interruttore generale e spia al neon a L. 74.000 Marino Severi - Piazza Isei, 28 - 47023 Cesena (FO) - Tel. 0547/20890

Vendo **riviste "Personal Software" dal nr. 2 al nr. 19** a L. 30.000; "M&P Computer" nr. 1/40 a L. 60.000; "Applicando" nr. 1/16 L. 45.000. Daniele Fonsato - Via Falconero-Villa Selenia - 10057 S.Ambrogio di Torino (TO) - Tel. 011/939417

Vendo **Sharp MZ-700 con stampante,** registratore, programmi, Assembler e 2 manuali a L. 700.000 trattabili. Roberto Mannini - Via A. Corelli, 10 - 50053 Empoli (FI) - Tel. 0571/91235

Iscriviti (gratuitamente) allo Sharp Club Monza. Cambio programmi Sharp, alcuni dei quali introvabili in Italia. Costantino Costanzi - Via Palestrina, 20 - 20052 Monza

Vendo **Spectravideo SV318 64 Kbyte quasi nuovo,** completo di joystick, cassette gioco e utilità a L. 450.000 trattabili. Telefonare ore serali. Alessandro Sconocchini - Via Tamburini, 21 - 60023 Ancona - Tel. 071/882396

Vendo **Apple II compatibile nuovo con monitor, 2 drive slim,** stampante, joystick; schede controller 80 colonne, CP/M, Z80, SAM Gen. mus., interfaccia per stampante, 100 floppy di programmi a L. 3.500.000 trattabili. Telefonare dopo le 20.00. Gerardo Franchini - Via Verdi, 25 - 38060 Nogaredo (TN) - Tel. 0464/412361

Vendo **Sharp MZ-721 con registratore e trasformatore incorporato,** manuale BASIC, 4 cassette giochi e cassetta linguaggio BASIC a L. 400.000. Telefonare ore serali. Stefano Baldi - Viale dei Salesiani, 20 - 00175 Roma - Tel. 06/744477

Vendo **Atari 130XE con soli 3 mesi di vita** più registratore Atari 1010, 4 cassette gioco e una cassetta "Invito a programmare"; il tutto in imballaggi originali a sole L. 500.000. Maurizio Cagneschi - Via Dei Cipressi, 6 - 10090 Bruino (TO) - Tel. 011/9086126

Vendo **Apple II compatibile con languade card incorporata,** 64 Kbyte, tastiera microprocessata maiuscole/minuscole a L. 450.000 trattabili. Oltre 150 programmi. Fabio Schettino - Via Emilia Ponente, 345 - 40100 Bologna - Tel. 051/569428

PERSONAL SOFTWARE **LIBRI** Gruppo Editoriale Jackson

Vuoi ordinare dei libri? Spedisci questo tagliando a:
Gruppo Editoriale Jackson
Via Rosellini, 12 – 20124 Milano

Nome Cognome

Indirizzo N.

Cap. Città Provincia

Partita I.V.A. (indispensabile per le aziende) Si richiede l'emissione della fattura ☐

Inviatemi i seguenti libri:

Codice Libro	Quantità	Codice Libro	Quantità	Codice Libro	Quantità	Codice Libro	Quantità

☐ Pagherò al postino il prezzo indicato + L. 3.000 per contributo fisso spese di spedizione.

☐ Allego assegno n° .... di L. ....

☐ Non abbonato ☐ Abbonato sconto ☐ l'Elettronica ☐ Elettronica Oggi ☐ Automazione Oggi ☐ Elektor ☐ Informatica Oggi
☐ Computerworld ☐ Bit ☐ Personal Software ☐ Strumenti Musicali ☐ Video Giochi ☐ Home computer ☐ Soft Dealer
☐ Telecomunicazioni Oggi ☐ PC Magazine ☐ Personal O ☐ Compuscuola ☐ Super VIC ☐ Super Commodore

Data .... Firma ....

PERSONAL SOFTWARE **PICCOLI ANNUNCI**

Sei un lettore di Personal Software e vuoi entrare in contatto con tutti gli altri lettori per comprare, cambiare o vendere il tuo sistema o qualche particolare accessorio? Spedisci questo tagliando a: Gruppo Editoriale Jackson - Via Rosellini, 12 - 20124 Milano

☐ COMPRO ☐ CAMBIO ☐ VENDO

Attenzione: Piccoli Annunci è aperta solamente alla vendita, l'acquisto o lo scambio di materiale fra privati, in un unico esemplare. Annunci di diversa natura verranno cestinati.

☐ Annuncio gratuito ☐ Annuncio evidenziato (allegare L. 5.000). Non si accettano fotocopie o fogli volanti allegati al coupon.

Nome .... Cognome ....

Via .... N. .... C.A.P. ....

Città .... Prov. .... Tel. ....

UNA PUBBLICAZIONE DEL GRUPPO EDITORIALE JACKSON

PERSONAL SOFTWARE

DIRETTORE RESPONSABILE: Giampietro Zanga

CAPO REDATTORE: Lorenzo Barrile

COORDINAMENTO TECNICO: Luca Zaninello

GRAFICA E IMPAGINAZIONE: Antonio Lanni

SEGRETARIA DI REDAZIONE: Silvia Borgonovo

UFFICIO ABBONAMENTI: Tel. (02) 6880951/2/3/4/5

HANNO COLLABORATO A QUESTO NUMERO:
V. Rossini, L. Accomazzi, S. Borsani,
F. Balena, M. Rogante, A. Nordio,
A. Zampariolo, M. Modelli, M. Giacobazzi

CONSOCIATE ESTERE:

USA: G.E.J. PUBLISHING Group, Inc. a Subsidiary of Gruppo Editoriale Jackson - 1307 S. Mary Avenue - 94087 Sunnyvale (CA) - Tel. (408) 7730103 - Telex: 49959972

GRAN BRETAGNA: G.E.J. Publishing L.t.d. - 187 Oxford Street - London W1R1AJ Tel. (01) 4392931 - Telex: (051) 21248

SPAGNA: Hispana S.P. - Calle Ortega y Gassat n. 25 28006 Madrid - Tel. (01) 2758600 - Telex (052) 22592

PUBBLICITA': Concessionario per l'Italia e l'Estero J. Advertising S.r.l. - Via Restelli, 5 - 20124 Milano Tel. (02) 6882895-6882458-6880606 - Telex: 316213 REINA I

PUBBLICITA' PER USA E CANADA:
International Media Marketing - 16704 Marquardt Avenue, P.O. Box 1217 Cerritos (CA) 90701 - (213) 926-9552

AGENTE PER LE TRE VENEZIE: Pierfilippo Obber - Via Nazionale, 70 38050 Imer (TN) - Tel. (0439) 67221.

FOTOCOMPOSIZIONE: Lineacomp S.r.l. - Via Rosellini, 12 - 20124 Milano

STAMPA: Litografica S.r.l. - (Busto Arsizio - MI)

CONCESSIONARIO ESCLUSIVO PER LA DIFFUSIONE IN ITALIA E ALL'ESTERO: SODIP - Via Zuretti, 25 - 20125 Milano

AUTORIZZAZIONE ALLA PUBBLICAZIONE:
Tribunale di Milano n. 69 del 20/2/1982

Pubblicità inferiore al 70%
Spedizione in abbonamento postale Gruppo III/70
Prezzo della rivista L. 4.500. Numero arretrato L. 9.000
Abbonamento annuo (11 numeri) L. 39.000; per l'Estero L. 78.000
I versamenti vanno indirizzati a: Gruppo Editoriale Jackson - Via Rosellini, 12 20124 Milano - mediante emissione di assegno bancario, cartolina vaglia o utilizzando il C/C postale numero 11666203.
Per i cambi di indirizzo, indicare, oltre naturalmente al nuovo, anche l'indirizzo precedente, ed allegare alla comunicazione l'importo di L. 500, anche in francobolli.

Il Gruppo Editoriale Jackson è iscritto nel Registro nazionale della Stampa al n. 117 - Vol. 2 - Foglio 129 in data 17-8-1982

GRUPPO EDITORIALE JACKSON
DIVISIONE PERIODICI
Milano-San Francisco-Londra-Madrid

Periodico iscritto all'Unione stampa periodica italiana

DIREZIONE, REDAZIONE, AMMINISTRAZIONE:
Via Rosellini, 12 - 20124 Milano - Telefoni: (02) 68.80.951/2/3/4/5
SEDE LEGALE: Via G. Pozzone, 5 - 20121 Milano
DIREZIONE EDITORIALE: Daniele Comboni
DIREZIONE AMMINISTRATIVA: Giuliano Di Chiano
DIREZIONE DIVISIONE LIBRI E GRANDI OPERE: Roberto Pancaldi
PIANIFICAZIONE STRATEGICA: Sergio Mello-Grand

per hobby e per professione

Il meglio di quel che c'è da leggere sulle Applicazioni Scolastiche del Computer!



250 volumi per approfondire, dalle più diverse angolazioni, le tematiche legate al mondo del computer e della microelettronica.

Per ordinare i volumi preferiti utilizzare l'apposito modulo di conto corrente postale blu o il coupon.

GRUPPO EDITORIALE JACKSON

R. THURNER
Programmazione strutturata
L'uso delle principali tecniche di rappresentazione strutturata, in analisi e programmazione. 136 pagine.
Cod. 503 A · L. 13.500 / *Abbonati L. 10.800*

R. FARABONE - R. VIANO
Logica e diagrammi a blocchi: tecniche di programmazione NOVITÀ
La diagrammazione a blocchi spiegata con chiarezza a chi programma. 400 pagine.
Cod. 539 A · L. 37.000 / *Abbonati L. 29.600*

M. ROSACLOT
La fisica con il computer
Le tecniche di simulazione sul calcolatore per studiare sistemi fisici. 268 pagine.
Cod. 550 A · L. 50.000. *Abbonati L. 40.000*
(con floppy per Apple II)

A.I.C.A.
Software didattico
L'informatica nella didattica: programmi disponibili, strumenti, prospettive. 408 pagine.
Cod. 804 C · L. 45.000 / *Abbonati L. 36.000*

G. TONFONI
La comunicazione cambiata
Come le nuove tecnologie modificano e potenziano le comunicazioni. 112 pagine.
Cod. 538 P · L. 10.500 / *Abbonati L. 8.400*

H. TAUB - D. SCHILLING
Elettronica integrata digitale
Un testo fondamentale con oltre 400 problemi per specialisti e studenti. 736 pagine.
Cod. 204 A · L. 45.000 / *Abbonati L. 36.000*

K.L. BOWIES
Soluzione di problemi con Pascal
Come imparare a programmare ed a risolvere problemi col calcolatore. 440 pagine.
Cod. 512 P · L. 28.000 / *Abbonati L. 22.400*

P. LE BEUX
Introduzione al BASIC
Un completo corso di BASIC, nella tecnica, nella pratica. 314 pagine.
Cod. 502 A · L. 25.000 / *Abbonati L. 20.000*

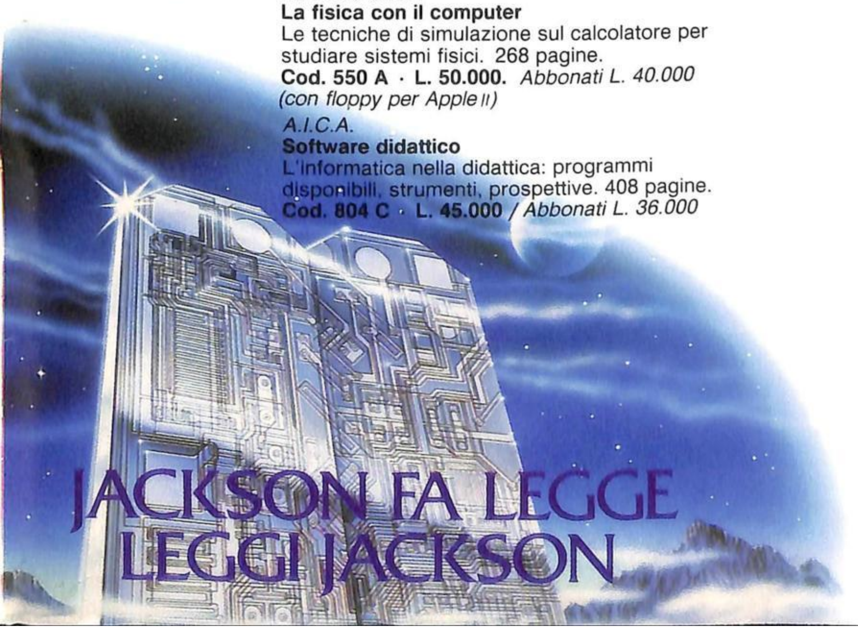
R. FARABONE - R. DORETTI
Dal Fortran IV al Fortran 77
Per programmare a livello tecnico-scientifico, imparando o approfondendo il Fortran. 276 pagine.
Cod. 517 P · L. 18.000 / *Abbonati L. 14.400*

J.P. LAMOITIER
50 esercizi in BASIC
Esercizi matematici, gestionali, statistici, per divertirsi, per imparare a programmare. 196 pagine.
Cod. 521 A · L. 13.000 / *Abbonati L. 10.400*

M. SANGIORGIO
Il Manuale del BASIC
Il BASIC di PC IBM, Apple, M20, MZ700, Spectrum, ZX81, C64, VIC 20, TI 99/4A a confronto per una rapida conversione. 486 pagine.
Cod. 534 A · L. 45.000 / *Abbonati L. 36.000*

J. SCRIVEN - P. HALL
Commodore 64 a scuola NOVITÀ
Come imparare, divertendosi, i segreti del C64. 184 pagine.
Cod. 574 D · L. 18.000 / *Abbonati L. 14.400*

E. DEESON
Lo Spectrum a scuola NOVITÀ
Per studenti e insegnanti di scuola media, una guida specifica allo Spectrum. Con cassetta. 282 pagine.
Cod. 579 D · L. 36.000 / *Abbonati L. 28.800*



JACKSON FA LEGGE
LEGGI JACKSON



È IN EDICOLA
VIDEO BASIC
PER DIALOGARE DAVVERO E SUBITO COL TUO COMPUTER
OGGI IN 5 VERSIONI
C=64
& C-128
MSX
C=16
VIC=20
Spectrum
IN REGALO UNA CASSETTA GIOCHI
GRUPPO EDITORIALE JACKSON
DIVISIONE GRANDI OPERE